Grant Number:	720270	Grant Title:	Human Brain Project SGA1

Deliverable Title:	D8.6.1 (D48.1 D14) SP8 Medical Informatics Platform – Architecture and Deployment Plan
Contractual Number and type:	SGA1 D8.6.1 (D48.1 D14) – Demonstrator
Dissemination Level:	PUBLIC
Version / Date:	Submitted : 04 June 2018; ACCEPTED 09 Jul 2018

Abstract:	This version of deliverable D8.6.1 describes the functional architecture of the Medical Informatics Platform at the end of the SGA1 project phase, including the overview of the MIP use case model and software component model. The document also describes the MIP product structure, including all MIP SGA1 components. The MIP data management concept is described, including data governance, data harmonisation and data privacy. Technology readiness levels, including the transition from ramp-up phase (RUP) to planned evolution during SGA2 phase, are thoroughly discussed. The technology readiness level of the MIP as an integrated system, as well as of the key components, has been assessed using the adapted TRL scale. Finally, the document gives a short overview of the deployed platforms in three European university hospitals and of a standard hospital deployment plan.
Keywords:	Cloud-ready, hybrid deployment model, community deployment, private hospital deployment, software-as-a-service, microservice architecture, Docker, Mezos, Marathon, harmonisation, brain scan, brain region volume, biomedical patient data, open research cohort, normalisation, de-normalisation, vector, tensor, distributed datasets, orchestration, descriptive statistics, inferential statistics, machine learning, gradient descent, neural networks, deep learning, analytical validity, clinical validity, clinical utility, data analytics

Targeted users/readers	Expert Reviewers – European Commission
Contributing Work-Package(s):	SGA1 WPs 8.1, 8.2, 8.3, 8.4, 8.5, 8.6
Initially Planned Delivery Date:	SGA1 M6 / 30 Sep 2016
History of changes:	This document is a completely new, restructured version of the document that was earlier submitted on 22 October 2016. The focus of the previous version was prospective – it provided a specification of the planned MIP SGA1 features, and a detailed project plan until SGA1 M24. The focus of the new document is retrospective – it provides the description of developed features, an overview of the use case model used to specify the developed functionality, a list of all delivered components, MIP data management strategy and artefacts, a technology readiness level assessment, including the previous ramp-up (RUP) phase and TRL projections for the next SGA2 project phase. Lastly, the tangible results in the form of the deployment of the platform in three European university hospitals are presented. Project planning is considered out of the scope of this document. The SGA1 project phase was finalised on 31 March 2018, and the proposal for the new SGA2 project phase was submitted to the European Commission. Instead of project planning, this document contains a discussion on the project end-results – developed components, supported functionality, technology readiness level and an overview of the deployed MIP instances in European hospitals.

Authors:	Dusan MILOVANOVIC, CHUV (P27)
Compiling Editors:	
Contributors:	Eva MIQEL FERNANDEZ, CHUV (P27)
SciTechCoord Review:	
Editorial Review:	EPFL (P1) Annemieke MICHELS

Table of Contents

List of Figures

List of Tables

1. Introduction

Convergence of biology and technology and the increasing capabilities to perform comprehensive “omic” assessments of an individual, including detailed brain features (morphology, connectivity, functionality), DNA sequence analysis, proteome, metabolome, microbiome, autoantibodies, physiome, phenome, etc., provide opportunities to discover new biological signatures of diseases, develop preventive strategies and improve medical treatments. Opportunities to use these data to improve health outcomes – to develop preventive strategies and improve medical care – are the motivation for the development of the Medical Informatics Platform (MIP).

The MIP is a cloud-ready patient data analysis ecosystem, which connects patient data from hospitals and research cohort datasets and provides a set of pre-integrated statistical methods and predictive machine learning algorithms for patient data exploration, data modelling, integration and execution of experiments (data analysis methods), and visualisation of the results.

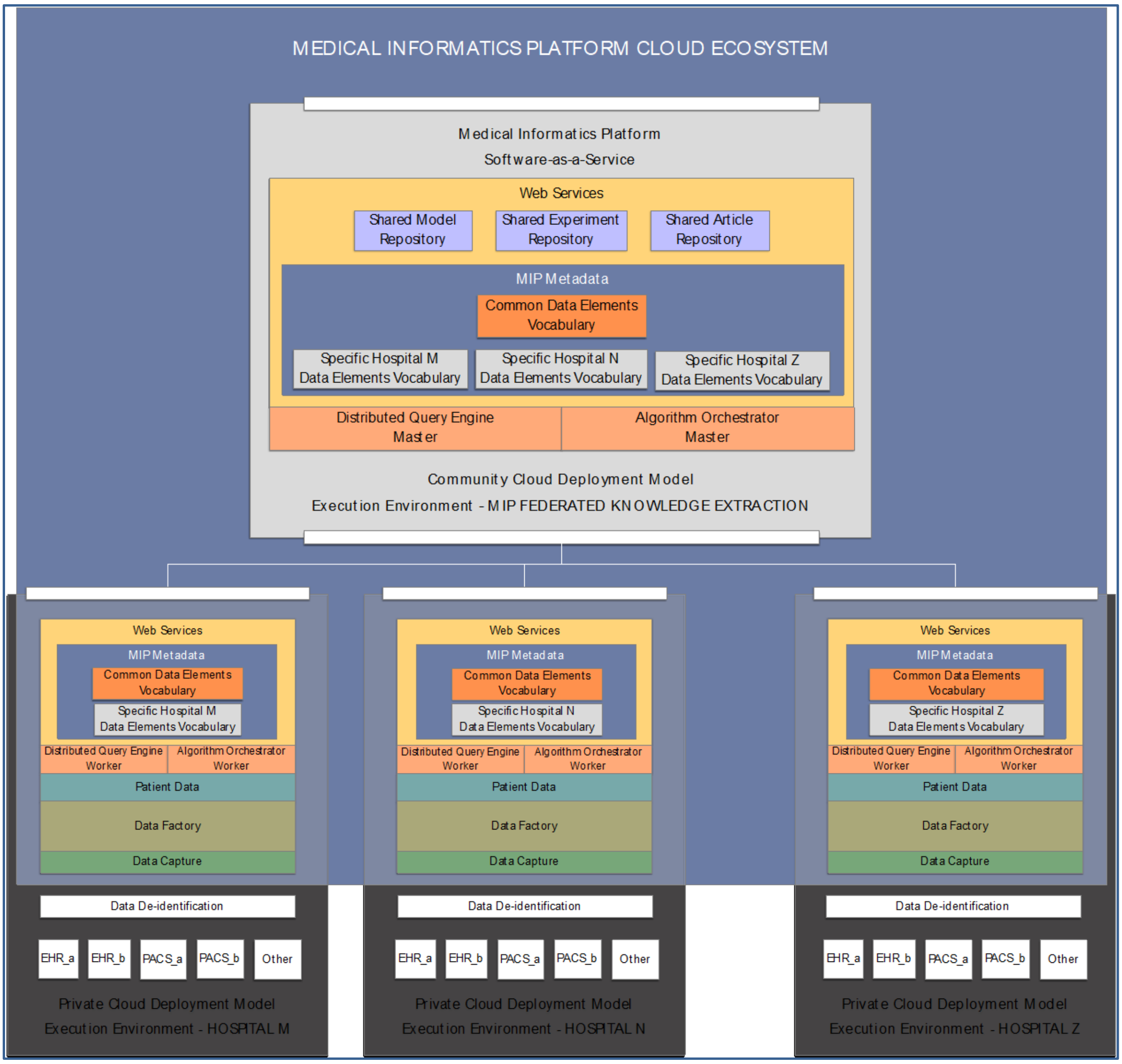

MEDICAL INFORMATICS PLATFORM CLOUD ECOSYSTEM
Medical Informatics Platform
Software-as-a-Service
Web Services
Shared Model Repository
Shared Experiment Repository
Shared Article Repository
MIP Metadata
Common Data Elements Vocabulary
Specific Hospital M Data Elements Vocabulary
Specific Hospital N Data Elements Vocabulary
Specific Hospital Z Data Elements Vocabulary
Distributed Query Engine Master
Algorithm Orchestrator Master
Community Cloud Deployment Model
Execution Environment - MIP FEDERATED KNOWLEDGE EXTRACTION
Web Services
MIP Metadata
Common Data Elements Vocabulary
Specific Hospital M Data Elements Vocabulary
Distributed Query Engine Worker
Algorithm Orchestrator Worker
Patient Data
Data Factory
Data Capture
Data De-identification
EHR_a
EHR_b
PACS_a
PACS_b
Other
Private Cloud Deployment Model
Execution Environment - HOSPITAL M
Web Services
MIP Metadata
Common Data Elements Vocabulary
Specific Hospital N Data Elements Vocabulary
Distributed Query Engine Worker
Algorithm Orchestrator Worker
Patient Data
Data Factory
Data Capture
Data De-identification
EHR_a
EHR_b
PACS_a
PACS_b
Other
Private Cloud Deployment Model
Execution Environment - HOSPITAL N
Web Services
MIP Metadata
Common Data Elements Vocabulary
Specific Hospital Z Data Elements Vocabulary
Distributed Query Engine Worker
Algorithm Orchestrator Worker
Patient Data
Data Factory
Data Capture
Data De-identification
EHR_a
EHR_b
PACS_a
PACS_b
Other
Private Cloud Deployment Model
Execution Environment - HOSPITAL Z

Figure 1: Medical Informatics Platform Architecture

The platform, developed during the SGA1 project phase, makes data on populations of patients broadly available for research use, by providing software-as-a-service to clinicians, neuroscientists and epidemiologists, for diagnosis and research in clinics, and for collaborative neuroscience research using hospital data and open patient research cohort datasets.

Figure 1 illustrates the cloud-ready MIP federated knowledge extraction software-as-a-service deployed in a community execution environment. It provides centralised access to the Medical Informatics Platform software and data deployed in private hospital execution environments. The MIP community execution environment orchestrates the execution of statistical and machine-learning algorithms for advanced multi-datasets, cross-centre descriptive, and predictive analytics and federates the results. The algorithms are executed locally, in private hospital execution environments where patient de-identified data are stored. Master orchestrator components that are running in community execution environment, connected to the distributed private MIP execution environments via web services, fetch the aggregated results of the algorithms executed in the private execution environments and aggregate them into a cross-centre data analysis result.

The MIP is engineered according to the privacy by design principle. De-identified patient data stored in private hospital execution environments are accessible only locally, either by the algorithms running there or by other means of data exploration within the private cloud using the locally deployed web services.

MIP users can access a community execution environment or a local private hospital execution environment through the MIP web portal. The MIP web applications allow statistical/aggregated (not individual) data exploration, selection of data types for analytics, execution of algorithms/experiments and visualisation of results. Figure 2 illustrates one instance of the web portal for the local execution environment in the University Hospital in Lausanne (CHUV), Switzerland.

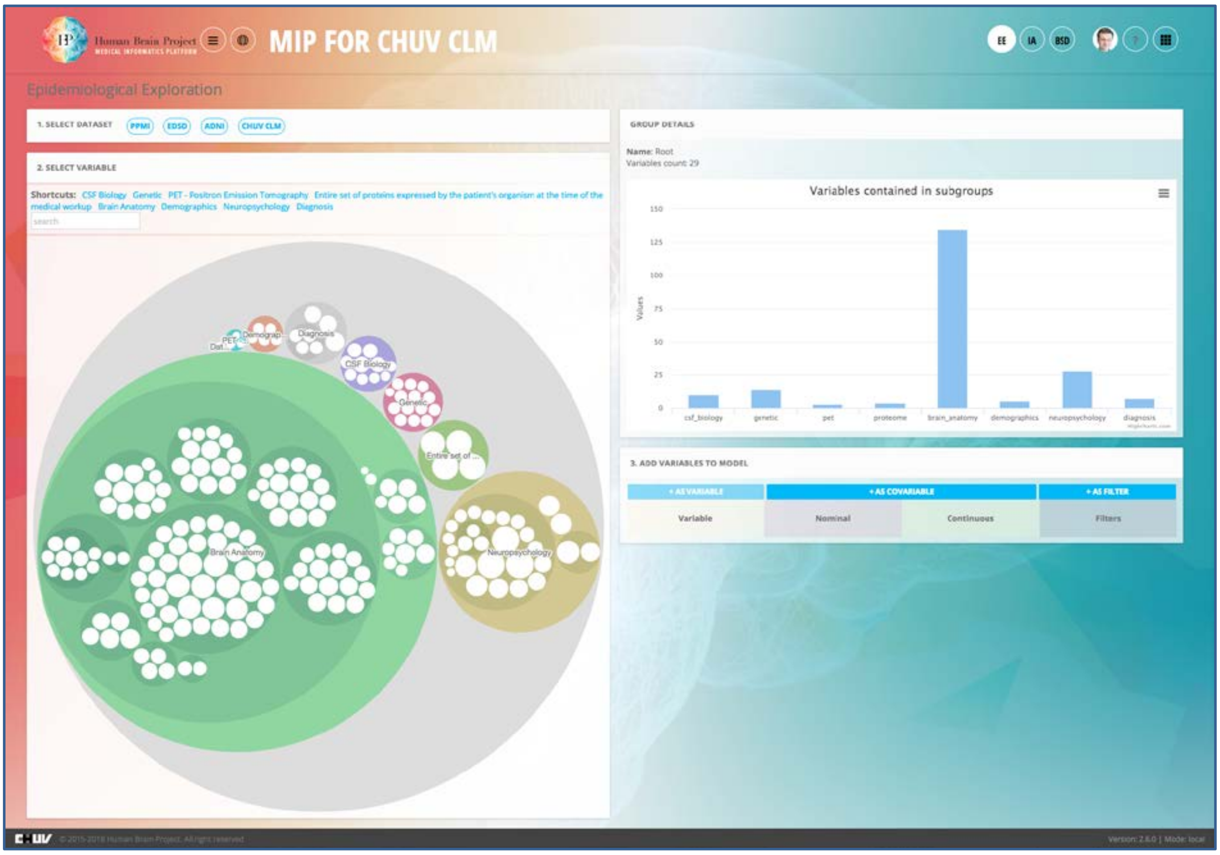

Human Brain Project
MEDICAL INFORMATICS PLATFORM
MIP FOR CHUV CLM
EE
IA
BSD
Epidemiological Exploration
1. SELECT DATASET
PPMI
EDSD
ADNI
CHUV CLM
2. SELECT VARIABLE
Shortcuts: CSF Biology Genetic PET - Positron Emission Tomography Entire set of proteins expressed by the patient's organism at the time of the medical workup Brain Anatomy Demographics Neuropsychology Diagnosis
search
PET
Demograp
Diagnosis
CSF Biology
Genetic
Entire set of
Brain Anatomy
Neuropsychology
GROUP DETAILS
Name: Root
Variables count: 29
Variables contained in subgroups
150
125
100
75
50
25
0
Values
csf_biology
genetic
pet
proteome
brain_anatomy
demographics
neuropsychology
diagnosis
3. ADD VARIABLES TO MODEL
+ AS VARIABLE
+ AS COVARIABLE
+ AS FILTER
Variable
Nominal
Continuous
Filters
CHUV
© 2015-2018 Human Brain Project. All right reserved
Version: 2.6.0 | Mode: local

Figure 2: Medical Informatics Platform Web Portal

2. Use Case Model

This section provides an overview of the Medical Informatics Platform use case model. Platform operational capabilities and user needs are formally defined using a use case modelling approach.

MIP use cases are identified using the traditional use case modelling approach: each use case specifies a complete functional unit, i.e. it handles the entire process, from its initiation by an external actor until it has performed the requested functionality. A use case always delivers some value to an actor.

There is a conceptual difference between MIP use cases and MIP use scenarios. MIP use cases represent a set of complete functions of the system, such as Data Exploration, Testing Correlation, Clinical Validity Assessment, etc. MIP use scenarios represent the workflows of the MIP functionalities to achieve a final result. Therefore, workflows of the MIP user scenarios, such as Measuring Clinical Utility of the Volumes of Medial Temporal Lobe Subregions for Diagnosing Alzheimer's Disease, consist of number of different MIP use cases used in a certain order and with a defined purpose.

This chapter gives an overview of the MIP use cases. Examples of MIP use scenarios are provided in SGA1 Deliverables D8.6.3 and D8.6.4.

2.1 Software Installation

The objective of this use case is to configure and install the Medical Informatics Platform software in a hospital's data centre.

The MIP microservices deployment architecture enables agile continuous integration and continuous component deployment developed or modified by different European-wide teams. This architecture enables efficient future upgrades of the platform with new technologies and new features needed to support evolved clinical needs. Automation of configuration and installation of the MIP software minimises IT efforts to keep the maximum focus on the scientific and clinical aspects of the projects.

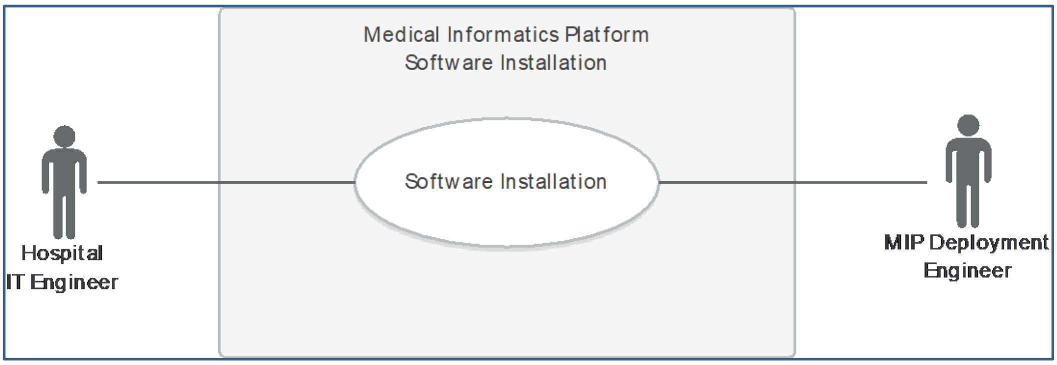


Figure 3: MIP Software Installation Use Case

<u>Scientific Added Value</u>

Hospital's data centre has a centralised platform for processing, storing and analysing de-identified and harmonised neuroimaging, neuropsychological, biological and demographic data of its patient population.

Efficient, configurable and automated end-to-end software installation, unifying operation system configuration, middleware installation and microservice building minimises the IT efforts to keep the focus on using the MIP platform for the scientific and clinical activities

2.2 Data Factory

The objective of the Data Factory use case group is to process patient data from different sources – hospitals and open research cohort datasets, EHR and PACS systems for:

1) Extraction of individual patient biomedical and health-related features
2) Transformation of source patient biomedical and health-related features to harmonised data structure and data vocabulary
3) Loading of transformed source datasets to permanent harmonised feature data store for federated multi-centre multi-dataset analytics

Patient source data from both hospitals and open research cohorts is typically structured and organised to capture the type and time of clinical observations, the type, modality, time and results of workups as well as the diagnoses. The Medical Informatics Platform is processing de-identified patient source data to extract biomedical and other health-related patient features, i.e. neuromorphometric, cognitive, biological, genetic, molecular and demographic, harmonises the extracted features across the different data sources, and permanently stores harmonised features for multi-centre, multi dataset clinical research studies.

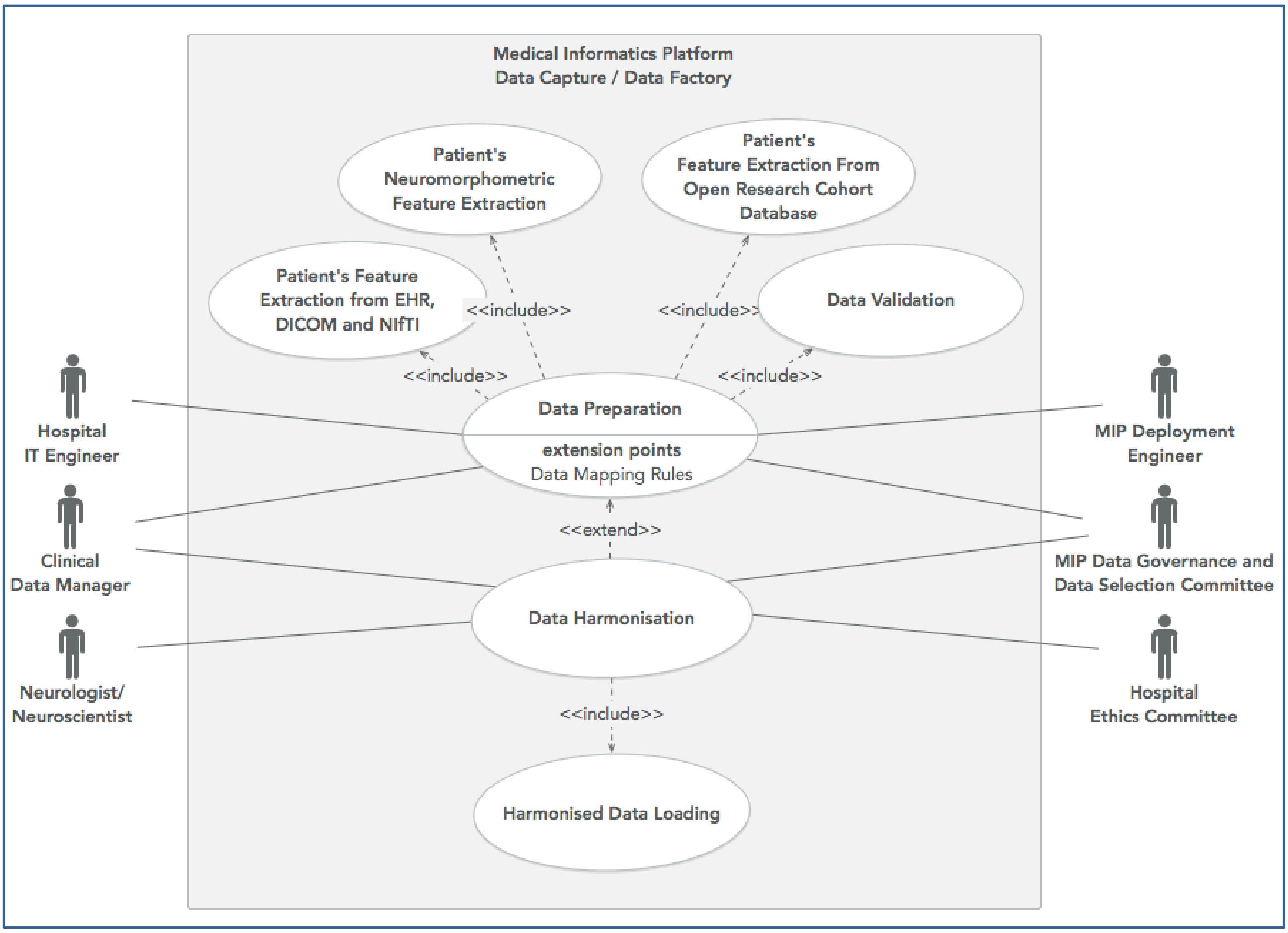

Medical Informatics Platform
Data Capture / Data Factory
Patient's Neuromorphometric Feature Extraction
Patient's Feature Extraction From Open Research Cohort Database
Patient's Feature Extraction from EHR, DICOM and NIfTI
Data Validation
<<include>>
<<include>>
<<include>>
<<include>>
Data Preparation
extension points
Data Mapping Rules
<<extend>>
Data Harmonisation
<<include>>
Harmonised Data Loading
Hospital IT Engineer
Clinical Data Manager
Neurologist/ Neuroscientist
MIP Deployment Engineer
MIP Data Governance and Data Selection Committee
Hospital Ethics Committee

Figure 4: MIP Data Factory Use Cases

Clinical studies involving multiple open research cohort datasets and patient datasets from multiple hospitals are challenging because data sources have different structures and use different coding systems. The Medical Informatics Platform supports harmonisation of data from different sources and provides harmonised data to clinicians and researchers for further analysis. This process is becoming more and more significant since the need for multi-centre studies is rapidly growing and the volume of the available open research cohort data have a tendency to explode.

Scientific Added Value

Extraction and harmonisation of patient biomedical and other health-related features from the source patient data is a first step in the process of creation of a data model for comprehensive molecular-level data analysis of both individual patients and populations, including their brain features, DNA sequence, proteome, metabolome, microbiome, autoantibodies, etc. Unification of biomedical and other health-related data provides the best opportunity to discover new biological signatures of diseases, improve taxonomy of diseases, develop preventive strategies, and improve medical treatment. This approach shall support the development of individualised medicine and enable cross-comparison between the individual patients to make diagnosing of complex cases more efficient and precise.

Harmonisation of the full set of Medical Informatics Platform's patient biomedical and other health-related features enables large multi-centre, multi-data source studies, increasing the accuracy of analysis methods and the probability for new scientific discoveries.

2.3 Web Applications

A web sub-system provides a web portal and the following applications:

- **Collaboration Space** – landing page of the Medical Informatics Platform displaying a summary of statistics (users, available variables, written articles), and the latest three shared models and articles. It provides a link to the Article Builder web application
- **Data Exploration** – a statistical exploration of patient feature data (i.e. variables). It is possible to explore only statistically aggregated data, not an individual patient's information. This web application provides on-the-fly generation of the descriptive statistics and contains a caching mechanism to handle any future data import in an automated way. It uses information stored in a Metadata database to display additional information about the displayed statistical data, such as data acquisition methodology, units, variable type (nominal or continuous), etc. This web application provides the functionality to search, select and classify data elements as variables, co-variables and filters for configuration of the statistical or machine learning models
- **Model Builder** – configuration/design of statistical or predictive machine learning models. It also provides visualisation for searching the data element types, select and classify data elements as variables, co-variables (nominal and continuous) and filters. Once the model is designed, a design matrix is populated with the selected data. Model Builder provides a visual representation of the design matrix and the selected data for inspection before running a statistical, feature extraction or a machine learning algorithms. It also provides an option to save the designed models
- **Model Validation –** measuring machine-learning models' accuracy by calculating predictive error rate of the model trained on training data against a test dataset. The results guide the user to select the best-performing algorithm and fine-tune its parameters as well as to understand how well the model performs before it is used. The Model benchmark and Validation component from Algorithm Factory is used to measure machine-learning model accuracy. In MIP SGA1 it supports cross-validation method – data split using K-Fold cross-validation. This method of data sampling divides the complete dataset into K disjoint parts of roughly the same size. K different models are trained on K-1 parts each while being tested on the remaining one part of the data. That is done on all K parts exactly once to ensure that every data row is used equally often for training and exactly once for testing. Resulting K test errors are then averaged to get the final error estimate of the model, which was built on the complete dataset
- **Experiment Builder & Disease Models** – a selection of a statistical, feature extraction or machine learning method, the configuration of the method's parameters and the parameters for the trained model validation for supervised machine learning, as well as launching of the

machine learning experiment. This application displays experiment validation results as bar charts and confusion matrices

- **Article Builder** – writing the articles using the results of the executed experiments
- **Third-party Applications and Viewers** – portal for accessing third-party web applications for data exploration and visualisation

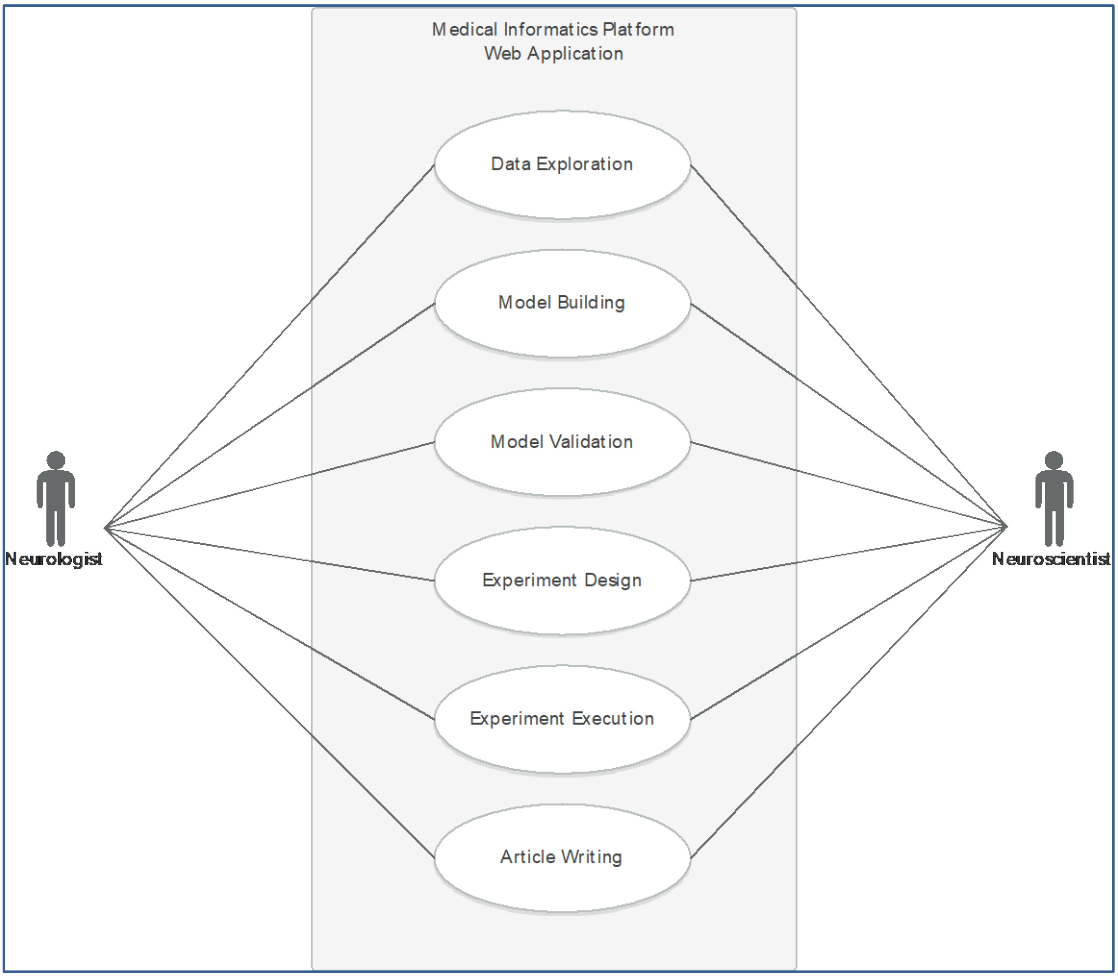


Figure 5: MIP Web Application Use Cases

2.4 Data Mining

The objective of data mining of a group of use cases is the discovery of properties of data in datasets. Out-of-the-box statistical and machine learning algorithms are used to realise MIP data mining use cases.

In case of using machine-learning algorithms for data mining, measurement of the learned model's accuracy and consequently the assessment of the accuracy of the discovered data properties is supported through using the algorithms from the Algorithm Factory's repository. Note that it is not possible to validate algorithms from the Distributed Query Processing Engine's repository in MIP SGA1.

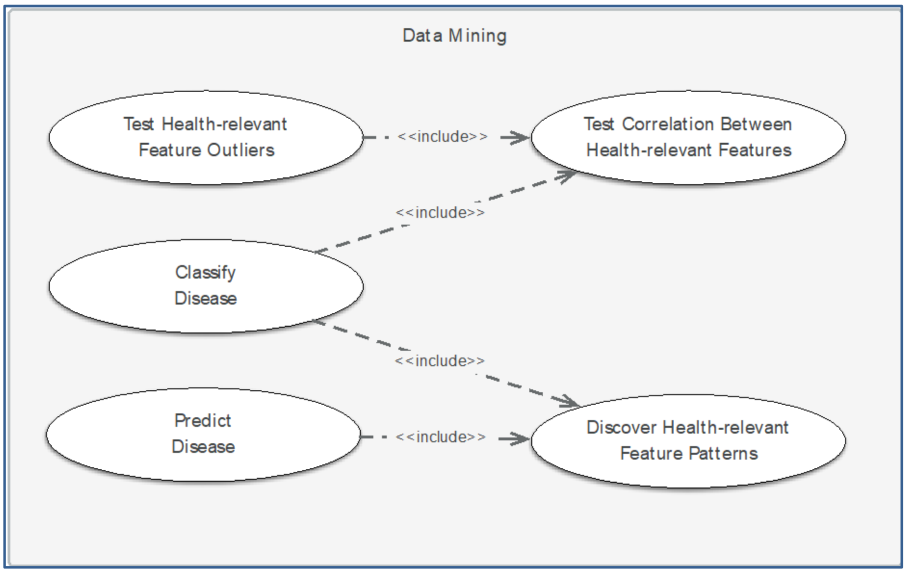

Data Mining
Test Health-relevant Feature Outliers
<<include>>
Test Correlation Between Health-relevant Features
<<include>>
Classify Disease
<<include>>
Predict Disease
<<include>>
Discover Health-relevant Feature Patterns

Figure 6: MIP Data Mining Use Cases

Scientific Added Value

This set of use cases specifies the core functionality of the MIP platform – data analytics. Any clinical / research operational scenario executes one or more of the data mining use cases. The four examples of scientific operational scenarios that execute all of the MIP data mining use cases are described in Chapter 8.

Example:

A correlation between brain volume and cognitive decline has been discovered. It was tested whether there are outliers: persons with brain volume decline but no cognitive decline. This gives the idea to include additional health-relevant features to discover whether they may correlate with the observed exceptions. For example, outliers have been discovered and with further data mining it was found that the age of the persons that have brain volume decline but no cognitive decline is in the same range – younger people who have brain volume decline do not have cognitive decline.

2.5 Data Analysis Accuracy Assessment

2.5.1 Analytical Validity

The MIP can be used to measure the analytical validity of tests, i.e. to measure the ability of the tests to accurately detect and measure patient health-related features of interest. MIP SGA1 can measure analytical validity of the following: brain MRI scans, scanning protocols, neuromorphometric feature extraction software applications, neuromorphometric feature extraction methods, neuropsychological instruments and methods, laboratory instruments and methods, etc.

The measured analytical validity using the MIP is the probability that the test results in a dataset chosen for the study will be in the same expected range with the results of the same test under the same conditions in different control datasets, i.e. other research cohorts with available data in the MIP. Analytical validity is a measurement of the MIP data quality.

When there are more data available in the MIP, meaning both the number of patients and the diversity of the test conditions and datasets, the measurement of analytical validity will be more accurate and reliable

The MIP can be used to measure analytical validity on its own, or to include measurement of analytical validity as a research dataset validation step prior to executing a scientifically relevant clinical or biomedical research study using that dataset.

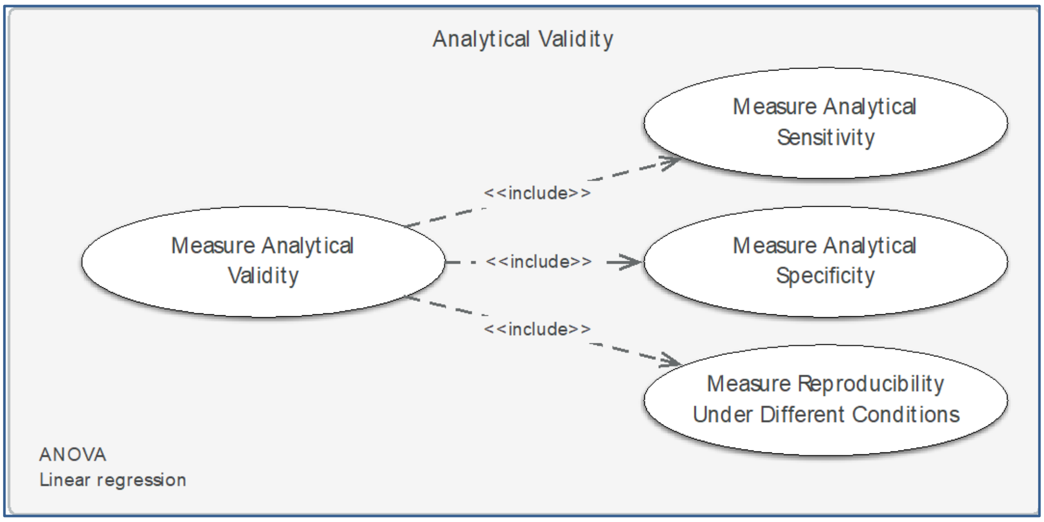

Analytical Validity
Measure Analytical Sensitivity
<<include>>
Measure Analytical Validity
<<include>>
Measure Analytical Specificity
<<include>>
Measure Reproducibility Under Different Conditions
ANOVA
Linear regression

Figure 7: Analytical Validity Use Case

Analytical validity is the test's ability to accurately detect and measure the biomarker of interest (i.e. protein, DNA, RNA). Are the test results repeatable when performed under identical conditions? Are the test results reproducible when the test is performed under different conditions? Is the test sensitive enough to detect biomarker levels as they occur in a real-life setting?

For DNA-based tests, analytical validity requires establishing the probability that a test will be positive when a particular sequence (analyte) is present (analytical sensitivity) and the probability that the test will be negative when the sequence is absent (analytical specificity). In contrast to DNA-based tests, enzyme and metabolite assays measure continuous variables (enzyme activity or metabolite concentration). One key measure of their analytical validity is accuracy, or the probability that the measured value will be within a predefined range of the true activity or concentration. Another measure of analytical validity is reliability, or the probability of repeatedly getting the same result.

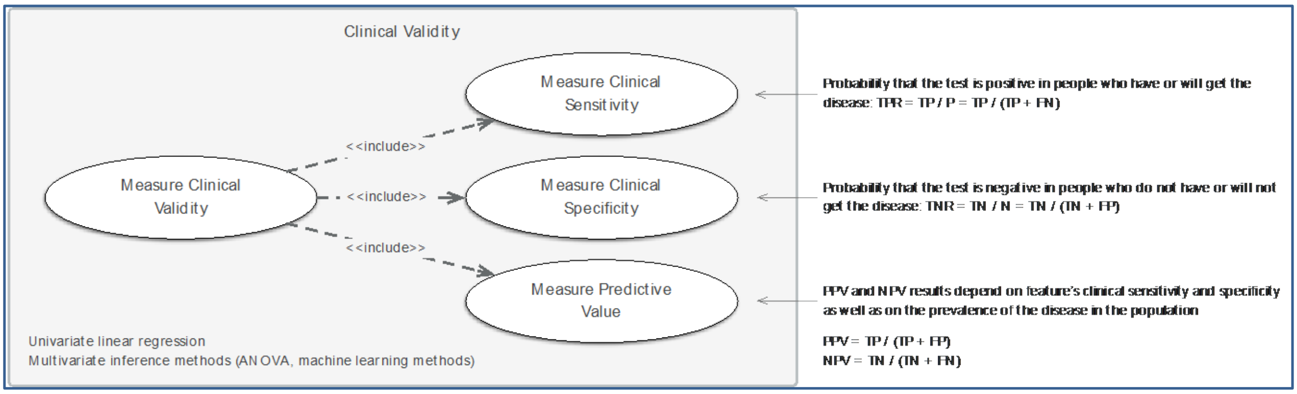
2.5.2 Clinical Validity

The MIP can be used to measure clinical validity of a biomarker or other health-relevant feature, i.e. to assess whether the biomarker or other health-relevant patient feature tested is associated with a disease or outcome or the response to a treatment.

Testing of whether a test is accurately detecting and measuring a biomarker or other health-relevant patient feature, i.e. the assessment of test's analytical validity, is a prerequisite for accurate and reliable measurement of the biomarker's or other health-relevant feature's clinical validity. To measure biomarkers' or other health-relevant features' clinical validity, the values for the tested biomarker or the other health-relevant feature, i.e. the data stored in MIP Feature Data Store, must be accurate and reliable. The MIP SGA1 can measure clinical validity of the following types of health-related features: neuromorphometric, cognitive, demographic, genetic, molecular and other biomedical metrics.

Assessment of clinical validity involves measurement of biomarker's or other health-relevant feature's clinical performance, including: (1) clinical sensitivity (ability to identify those who have or will get the disease), (2) clinical specificity (ability to identify those who do not have or will not get the disease), (3) positive predictive value (PPV) - the probability that a person with a positive test result for a predictor, i.e. a biomarker or other health-relevant feature, has or will get the disease, and negative predictive value (NPV) - the probability that a person with a negative test result for a predictor, i.e. a biomarker or other health-relevant feature, does not have or will not get the disease.

When there are more data available in MIP, meaning the number of patients and the diversity of their conditions and profiles, the measurement of clinical validity will be more accurate and reliable.


Clinical Validity
Measure Clinical Validity
<<include>>
<<include>>
<<include>>
Measure Clinical Sensitivity
Measure Clinical Specificity
Measure Predictive Value
Univariate linear regression
Multivariate inference methods (ANOVA, machine learning methods)

Probability that the test is positive in people who have or will get the disease: TPR = TP / P = TP / (TP + FN)

Probability that the test is negative in people who do not have or will not get the disease: TNR = TN / N = TN / (TN + FP)

PPV and NPV results depend on feature's clinical sensitivity and specificity as well as on the prevalence of the disease in the population

PPV = TP / (TP + FP)
NPV = TN / (TN + FN)

Figure 8: Clinical Validity Use Case

MIP can be used to measure clinical validity on its own, or to include measurement of clinical validity as a research dataset validation step prior to executing a scientifically relevant clinical or biomedical research study using that dataset.

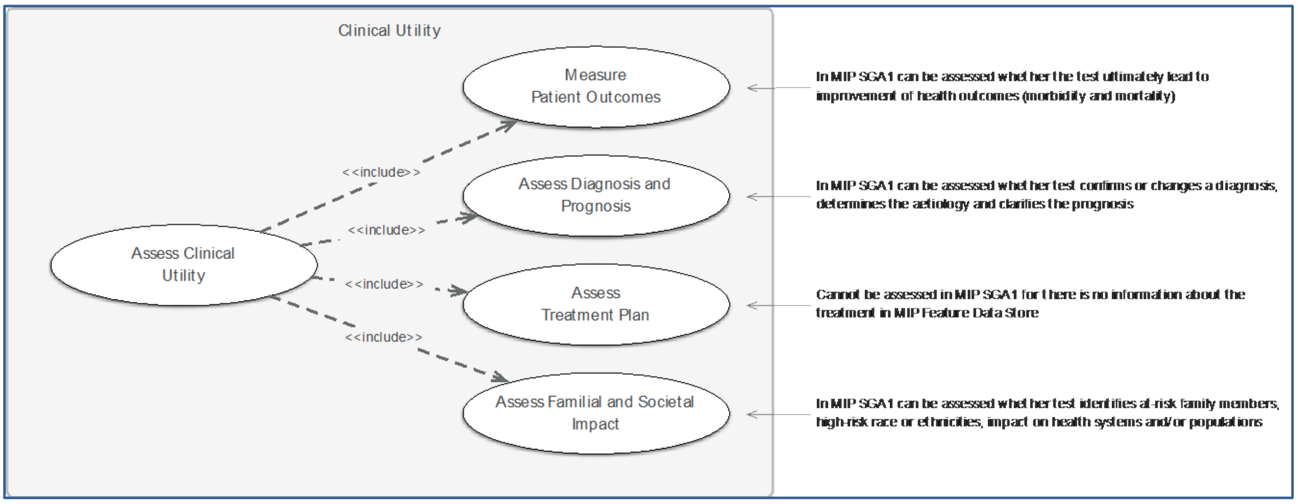
2.5.3 Clinical Utility

Clinical utility is perhaps one of most important considerations when determining whether or not to order or cover a biomedical or other health-relevant feature test. While the meaning of the term has some variability depending on the context or source, there is a largely agreed-upon definition. Four factors are generally considered when evaluating the clinical utility of a test:

- **Patient outcomes** – do the results of the test ultimately lead to improvement of health outcomes (e.g. reduce mortality or morbidity) or other outcomes that are important to patients such as quality of life?
- **Diagnostic thinking** – does the test confirm or change a diagnosis? Does it determine the aetiology for a condition or does it clarify the prognosis?
- **Decision-making guidance** – will the test results determine the appropriate dietary, physiological, medical (including pharmaceutical), and/or surgical intervention?
- **Familial and societal impacts** – does the test identify family members at risk, high-risk race/ethnicities, and the impact on health systems and/or populations?

The development of tests to predict future disease often precedes the development of interventions to prevent, ameliorate, or cure that disease. Even during this therapeutic gap, benefits might accrue from testing. However, in the absence of definitive interventions for improving outcomes in those with positive test results, the clinical utility of the testing will be limited. To improve the benefits of testing, efforts must be made to investigate the safety and effectiveness of new interventions while the tests are developed.

Clinical utility is not always evident in testing for inherited disorders for which treatments have not yet been developed. The clinical utility of a genetic diagnosis for an incurable or untreatable disease, without knowing the outcome, just looking for a predisposition to disease, is not useful.



In MIP SGA1 can be assessed whether the test ultimately lead to improvement of health outcomes (morbidity and mortality)

In MIP SGA1 can be assessed whether test confirms or changes a diagnosis, determines the aetiology and clarifies the prognosis

Cannot be assessed in MIP SGA1 for there is no information about the treatment in MIP Feature Data Store

In MIP SGA1 can be assessed whether test identifies at-risk family members, high-risk race or ethnicities, impact on health systems and/or populations

Figure 9: Clinical Utility Use Case

2.6 Overview of MIP Use Cases

This section of the document gives a summary of all MIP use cases discussed in the previous subchapters. MIP use cases are grouped in two tables:

1) MIP use cases involved in MIP deployment use scenarios

2) MIP use cases involved in MIP data analysis, i.e. clinical study scenarios

MIP deployment use scenarios consist of the actions for software installation and patient data extraction and processing.

MIP clinical study scenarios consist of data analysis actions, including data examination, creation of data models, selection and configuration of statistical or machine learning methods for descriptive or predictive data analytics.

Table 1: Overview of MIP Deployment Use Cases

Medical Informatics Platform Use Case List			
ID	**Name**	**Short Description**	**Relationship**
Deployment Use Cases			
Software Installation			
UC_ITL_01	Software Installation	MIP execution environment configuration and software installation	
Data Capture / Data Factory			
UC_DFY_01	Data Preparation	Orchestration of source EHR and brain imaging data extraction, data transformation and data loading pipelines, including data quality assurance and data provenance storage	

Medical Informatics Platform Use Case List			
ID	**Name**	**Short Description**	**Relationship**
Deployment Use Cases			
UC_DFY_02	Patient's Feature Extraction from EHR, DICOM and NIfTI	Extraction of patient demographic, biological, genetic and cognitive data from HER and extraction of the metadata from patient's brain scan DICOM or NIfTI files	Included in UC_DFY_01
UC_DFY_03	Patient's Neuromorphometric Feature Extraction	Extraction of neuromorphometric data from patient brain scans	Included in UC_DFY_01
UC_DFY_04	Patient's Feature Extraction From Open Research Cohort Dataset	Extraction of patient feature data from open research cohort datasets	Included in UC_DFY_01
UC_DFY_05	Data Validation	Checking of pre-processed brain images for artefacts and quality metrics, check data for confound and biases, check metadata	Included in UC_DFY_01
UC_DFY_06	Data Harmonisation	Transformation of source patient biomedical and health-related features to harmonised data structure and data vocabulary	Extends UC_DFY_01
UC_DFY_07	Harmonised Data Loading	Loading of transformed source datasets to permanent harmonised feature data store for federated multi-centre multi-dataset analytics	Included in UC_DFY_05

Table 2: Overview of MIP Clinical Study Use Cases

Medical Informatics Platform Use Case List			
ID	**Name**	**Short Description**	**Relationship**
Clinical Study Use Cases			
Web Application			
UC_WEB_01	Data Exploration	Statistical exploration of patient feature data (i.e. variables)	

Medical Informatics Platform Use Case List			
ID	**Name**	**Short Description**	**Relationship**
Clinical Study Use Cases			
UC_WEB_02	Model Building	Configuration/design of statistical or predictive machine learning models	
UC_WEB_03	Model Validation	Validation of learned model against the test dataset. Calculation of the predictive error rate	
UC_WEB_04	Experiment Design	Selection of a statistical, feature extraction or machine learning method, the configuration of the method's parameters and the parameters for the trained model validation for supervised machine learning	
UC_WEB_05	Experiment Execution	Launching of the machine learning experiment. Displays experiment validation results as bar charts and confusion matrices	
UC_WEB_06	Article Writing	Writing scientific articles using the results of the executed experiments	
Data Mining			
UC_DTM_01	Test Correlation Between Health-relevant Features	Testing the correlation between two or more variables using a statistical or machine learning method	Included in UC_DTM_02 Included in UC_DTM_03
UC_DTM_02	Test Health-relevant Feature Outliers	Discovering outliers after testing the correlation between variables	
UC_DTM_03	Classify Disease	Using classification machine learning algorithms to create (learn), validate and/or apply the classifier	
UC_DTM_04	Predict Disease	Apply a learned classifier to predict pathology	
UC_DTM_05	Discover Health-relevant Feature Patterns	Discover patterns of correlated variables in a population	Included in UC_DTM_03 Included in UC_DTM_04
Data Analysis Accuracy Assessment			

Medical Informatics Platform Use Case List			
ID	**Name**	**Short Description**	**Relationship**
Clinical Study Use Cases			
UC_ACC_01	Measure Biomarker's Analytical Validity	Measure analytical validity of tests – assess the ability of the test to accurately detect and measure patient's health-related features of interest. Analytical validity measured using MIP is the probability that the test results in a dataset chosen for the study will be in the same expected range with the results of the same test under the same conditions in different control datasets, i.e. other research cohorts whose data are part of the MIP. Analytical validity is a measurement of the MIP data quality.	
UC_ACC_02	Measure Biomarker's Analytical Sensitivity	Measure the probability that a test will detect an analyte when it is present in a specimen	Included in UC_ACC_01
UC_ACC_03	Measure Biomarker's Analytical Specificity	Measure the probability that a test will be negative when an analyte is absent from a specimen	Included in UC_ACC_01
UC_ACC_04	Measure Biomarker's Reproducibility Under Different Conditions	Evaluating the results of the a test when it is performed under different conditions	Included in UC_ACC_01
UC_ACC_05	Measure Health-relevant Feature's Clinical Validity	Measure clinical validity of a biomarker or other health-relevant feature, i.e. to assess whether the biomarker or other tested health-relevant patient's feature is associated with a disease or outcome or the response to a treatment	
UC_ACC_06	Measure Health-relevant Feature's Clinical Sensitivity	Probability that the test is positive in people who have or will get the disease: TPR = TP / P = TP / (TP + FN)	Included in UC_ACC_05
UC_ACC_07	Measure Health-relevant Feature's Clinical Specificity	Probability that the test is negative in people who do not have or will not get the disease: TNR = TN / N = TN / (TN + FP)	Included in UC_ACC_05

Medical Informatics Platform Use Case List			
ID	**Name**	**Short Description**	**Relationship**
Clinical Study Use Cases			

UC_ACC_08	Measure Health-relevant Feature's Clinical Predictive Value	Positive Predictive Value (PPV) and Negative Predictive Value (NPV) results depend on feature’s clinical sensitivity and specificity as well as on the prevalence of the disease in the population.
		PPV = TP / (TP + FP)
		NPV = TN / (TN + FN)
			Included in UC_ACC_05

Clinical Utility Assessment			
UC_CLU_01	Assess Health-relevant Feature's Clinical Utility	Three factors are generally considered when evaluating the clinical utility of a test: 1) Patient outcomes, 2) Diagnostic thinking, 3) Societal impacts	
UC_CLU_02	Measure Patient Outcomes	Do the results of the test ultimately lead to improvement of health outcomes (e.g. reduced mortality or morbidity) or other outcomes that are important to patients such as quality of life?	Included in UC_CLU_01
UC_CLU_03	Assess Diagnosis and Prognosis	Does the test confirm or change a diagnosis? Does it determine the aetiology for a condition or does it clarify the prognosis?	Included in UC_CLU_01
UC_CLU_04	Assess Societal Impact	Does the test identify high-risk race/ethnicities, and the impact on health systems and/or populations?	Included in UC_CLU_01

2.7 Data Analysis Accuracy Assessment Use Case Overview

The assessment of the accuracy of analysed patient data – the MIP diagnostic measure – is a crucial feature of any clinical study. As discussed in previous subchapters, an assessment of data analysis results is performed by the three use case groups: Analytical Validity, Clinical Validity and Clinical Utility. Here, we provide an overview of the three groups, the object of measurement, a short description and a list of statistical and machine learning methods available in the MIP.

Table 3: Measuring Analytical and Clinical Validity and Clinical Utility With The MIP

MIP Diagnostic Measure	Measured Object	Description of the MIP Diagnostic Measure	Method of Measure
Analytical Validity	**Data Quality**	Measurement of the data quality: accuracy and reliability. Accuracy is the probability that the values of the patient features in a dataset chosen for the study will be in the same expected range as the values of those features in "gold standard" – control research cohort datasets. Reliability is the probability of repeatedly getting the same data analysis result when using MIP's integrated statistical methods and machine learning algorithms. Analytical validity assessment is a prerequisite for accurate and reliable measurement of the feature's clinical validity. To measure clinical validity of the feature, data stored in MIP must be accurate and reliable Reliability of the predictive (machine-learning) models is measured using model validation methods integrated into the Medical Informatics Platform	Analytical validity of data: ANOVA, linear regression, logistic regression. Visual methods: histogram, density plot, scatter plot, box plot Analytical validity of predictive models: cross-validation
Clinical Validity	**Clinical Feature**	Measurement of the feature's clinical performance: (1) clinical sensitivity (ability to identify those who have or will get the disease), (2) clinical specificity (ability to identify those who do not have or will not get the disease), (3) positive predictive value (PPV, the probability that a person with a positive test result for a predictor, has or will get the disease), and negative predictive value (NPV the probability that a person with a negative test result for a predictor does not have or will not get the disease). The MIP can be used to measure clinical validity of the features (biomarkers and other relevant data), or to measure clinical validity of the descriptive and predictive mathematical models by executing integrated model validation methods. Clinical validity of the models with different set of features can be compared using ROC curves, C-statistics, etc.	Clinical validity of features: ANOVA, linear regression, logistic regression. Visualisation: heatmap

MIP Diagnostic Measure	Measured Object	Description of the MIP Diagnostic Measure	Method of Measure
		The more data available in the MIP (patient number and diversity of patient conditions and profiles), the more accurate and reliable the measurement of clinical validity	
Clinical Utility	Result of Analytics	Evaluation of the clinical utility of the results of the data analytics using the Medical Informatics Platform: 1) **diagnostic relevance**: do the results of the predictive analytics confirm or change a diagnosis in a new group of patients, do they determine the aetiology for a condition or clarify the prognosis 2) **disease outcomes**: do the results of the predictive analytics lead to the improvement of health outcomes (e.g. reduce mortality or morbidity – prescriptive implication of machine learning models) or other outcomes that are important to patients, such as quality of life 3) **familial and societal impacts**: do the results of the predictive analytics identify family members at risk, high-risk race/ethnicities, and the impact on health systems and/or population The important part of the assessment of the clinical utility of the results of predictive analytics is the evaluation of the accuracy of the hypothesis function. The method used in this release of MIP is cross-validation. The measured accuracy of the learned model shall determine the level of clinical utility of the model with the real patient population.	Machine learning models (supervised and unsupervised): univariate and multivariate linear and polynomial regression using gradient decent, KNN, Naïve Bayes; K-means; SVM. Validation of machine learning models using cross-validation integrated into the MIP

3. MIP Architecture

The Medical Informatics Platform is a complex information system comprising numerous software components designed and integrated by different SP8 partners.

This Chapter provides an end-to-end functional overview of the Platform, describing the logical component architecture and the components' roles, showing how the functionality is designed inside the Platform, regarding the static structure of the Platform and the interaction between its components.

This Chapter also contains a brief overview of the key deployment architecture concepts, without providing a detailed specification of the deployment of components into the Platform's physical architecture. Some deployment terminology, such as "local hospital MIP" and "central MIP federation node" is used here only in the context of describing the function of relevant components.

3.1 Functional Architecture Overview

3.1.1 Data Capture Subsystem

The Data Capture sub-system provides a local interface to other hospital information systems. It is a single point of entry for all the data that contain personally identifiable information.

The purpose of the Data Capture sub-system is de-identification of patient data exported from hospital information systems (EHRs, PACS). De-identified data is uploaded to De-identified Data Version Control Storage, belonging to the Data Factory sub-system, for processing and feature extraction.

The flow of data between the Data Capture component (Data De-identifier) and, on one side, other local hospital information systems and, on the other side, the MIP Data Factory sub-system is as follows:

1) MIP captures personal health sensitive data from the following hospital information systems:
    - Electronic Health Record (EHR) Systems
    - Picture Archiving and Communication Systems (PACS)
2) Data De-identifier replaces the following personally identifiable information with pseudonyms:
    - Information exported from EHR systems in CSV format
    - Information from neuroimages stored in the headers of DICOM files
3) Data De-identifier saves the files with de-identified data to storage in the Data Factory sub-system

Anonymised patient cohort datasets (for example, ADNI, EDSD, PPMI) are stored directly in the De-identified Data Version Controlled Storage belonging to the Data Factory sub-system.

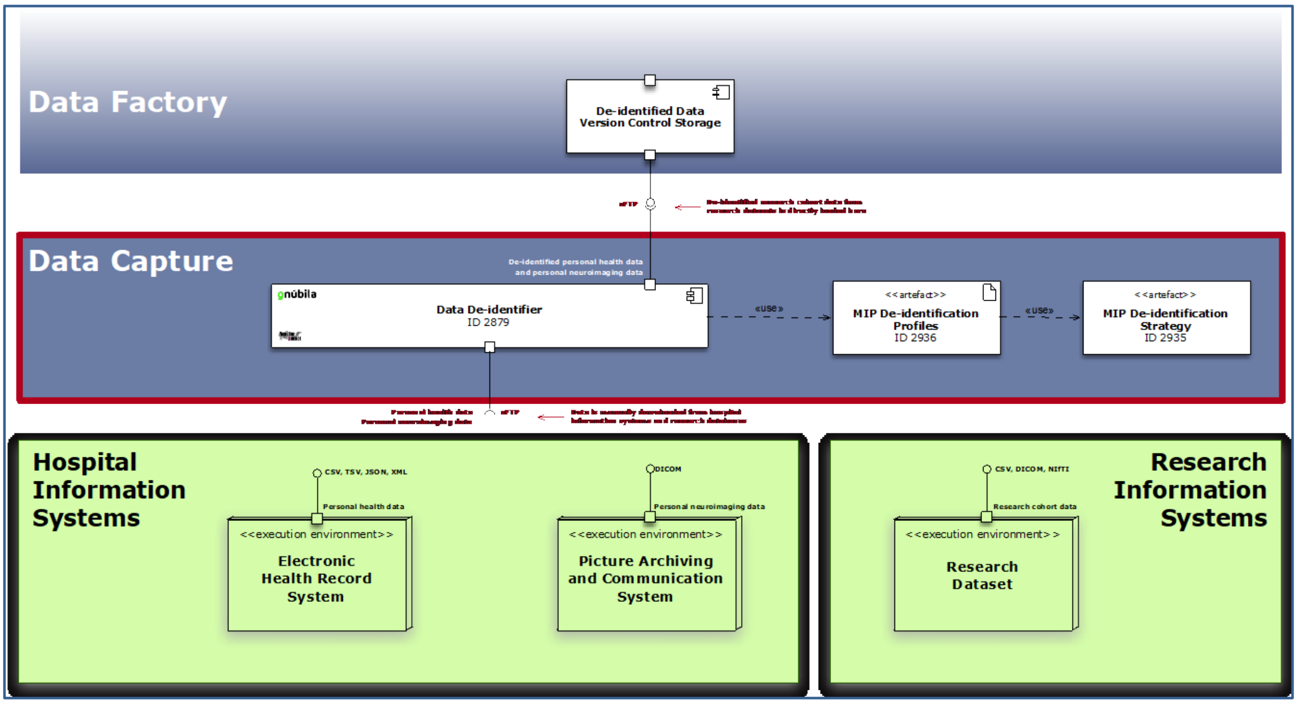
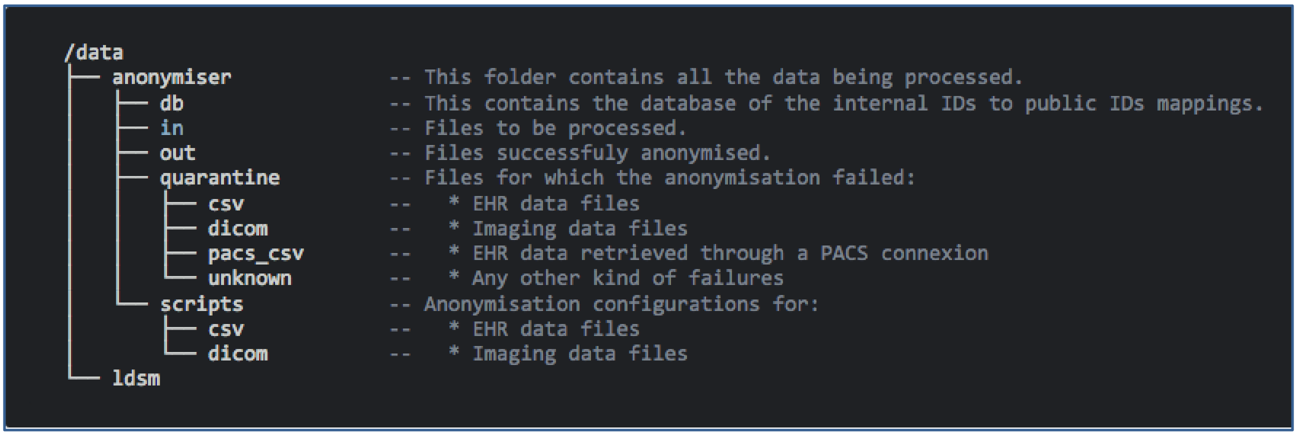

Data Factory
De-identified Data Version Control Storage
sFTP
De-identified research cohort data from research datasets is directly loaded here
Data Capture
De-identified personal health data and personal neuroimaging data
gnúbila
Data De-identifier
ID 2879
«use»
<<artefact>>
MIP De-identification Profiles
ID 2936
«use»
<<artefact>>
MIP De-identification Strategy
ID 2935
Personal health data
Personal neuroimaging data
sFTP
Data is manually downloaded from hospital information systems and research databases
Hospital Information Systems
CSV, TSV, JSON, XML
Personal health data
<<execution environment>>
Electronic Health Record System
DICOM
Personal neuroimaging data
<<execution environment>>
Picture Archiving and Communication System
Research Information Systems
CSV, DICOM, NIfTI
Research cohort data
<<execution environment>>
Research Dataset

Figure 10: Data Capture Sub-system



```
/data
├── anonymiser                -- This folder contains all the data being processed.
│   ├── db                    -- This contains the database of the internal IDs to public IDs mappings.
│   ├── in                    -- Files to be processed.
│   ├── out                   -- Files successfuly anonymised.
│   ├── quarantine            -- Files for which the anonymisation failed:
│   │   ├── csv               --   * EHR data files
│   │   ├── dicom             --   * Imaging data files
│   │   ├── pacs_csv          --   * EHR data retrieved through a PACS connexion
│   │   └── unknown           --   * Any other kind of failures
│   └── scripts               -- Anonymisation configurations for:
│       ├── csv               --   * EHR data files
│       └── dicom             --   * Imaging data files
└── ldsm
```

Figure 11: Data Folder Organisation for the De-identification Processing

The Electronic Health Record (EHR) is a collection of a patient health information stored by EHR systems in a digital format. EHR systems are designed for capturing and storing of patient data over time. Well-designed EHR systems are online transaction processing systems that collect and store patient data in a normalised database, therefore minimising data redundancy and improving data integrity.

Picture Archiving and Communication System (PACS) provides storage and access to digital images originating from multiple modalities (imaging machine types). The universal format for PACS image storage and transfer is DICOM (Digital Imaging and Communications in Medicine). Non-image data, such as image-related metadata and scanned PDF documents, can be encapsulated in DICOM files.

MIP captures patient personally identifiable demographic, diagnostic and biomedical data from EHR systems in CSV file format and neuroimaging MRI data from PACS systems in DICOM file format. Patient data are captured periodically for batch processing in the MIP.

Authorised hospital staff that exported the data, manually imports them into the MIP Data De-identifier component for de-identification.

In coordination with local hospital's data management team and ethics committee, the MIP data governance and data selection team (DGDS) is responsible for the specification of data depersonalisation rules in compliance with data protection regulations, such as EU/GDPR, CH/FADP and US/HIPAA. The Data de-identifier component's rule engine is configured using configuration scripts derived from these rules.

The third-party Gnubila FedEHR Anonymizer data de-identification solution has been chosen for the Data De-identifier component. This component is a profile-based, rule-based asynchronous message-oriented mediation engine, developed using an Apache Camel framework. It can be extended to support new data formats and de-identification algorithms. It replaces all personally identifiable information from the captured data with pseudonyms using out-of-the-box data de-identification techniques, such as generalisation, micro-aggregation, encryption, swapping and sub-sampling.

Discussion About Data Re-identification

Data re-identification is not a feature of the Medical Informatics Platform. It is not possible to re-identify a patient using any of the designed functions of the MIP (data privacy by design). Administratively and organisationally, re-identification of patient data is the responsibility of their hospitals. Technically, for re-identifying patient data stored in the de-identified form in their hospitals' local MIP data storage, hospital IT staff needs to develop standalone lookup applications to map personally identifiable information with the pseudonyms at the point of de-identification. Those applications shall never be integrated with the MIP.

3.1.2 Data Factory Subsystem

The components of the logical Data Factory sub-system perform batch neuroimaging and EHR data pre-processing, extraction, transformation and loading into the normalised permanent data storage.

The ETL processes of the Data Factory sub-system are orchestrated as directed acyclic graphs (DAG's) of tasks in programmatically configurable pipelines using an open-source Apache Airflow workflow management platform. Additional components are built for data transformation and data provenance tracking, including the complex neuroimaging processing and brain feature extraction, brain scan metadata and EHR data extraction as well as data transformation and loading tasks.

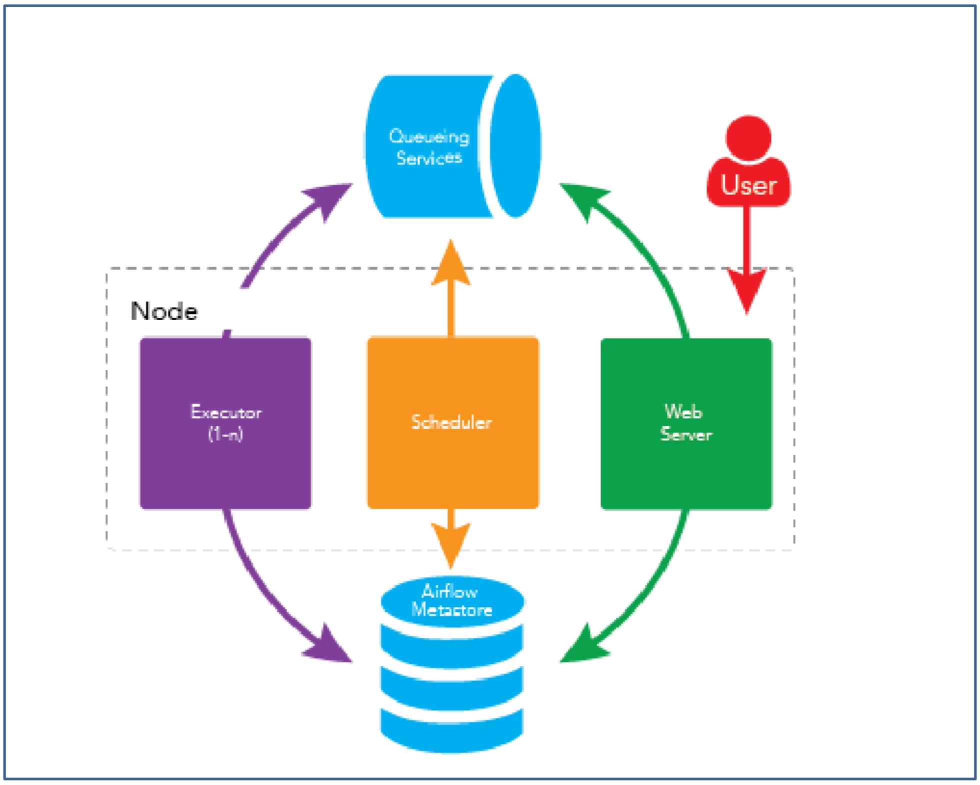


Figure 12: Apache Airflow Concept

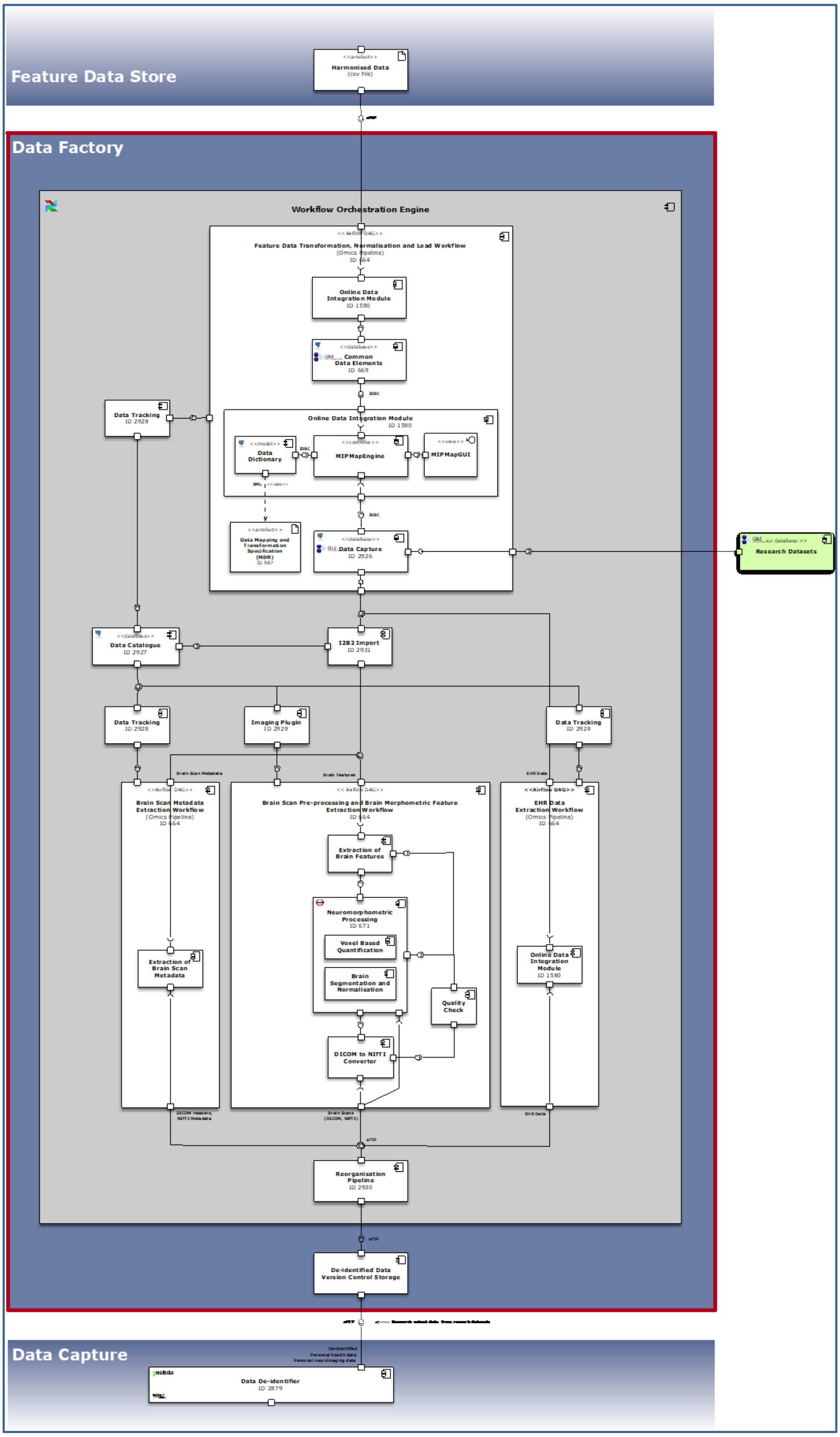
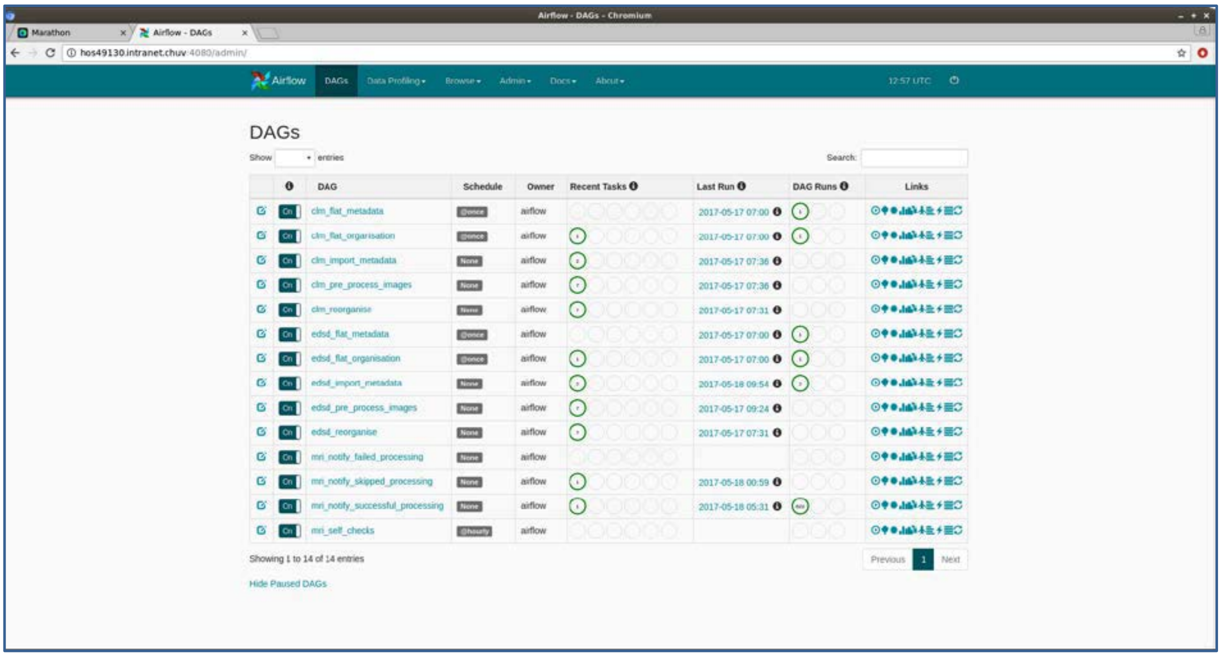


Figure 13: Data Factory Sub-system

Airflow is an open source solution for defining, scheduling, and monitoring of jobs. Pipelines are defined as a code using Python and the jobs are scheduled using cron expressions. The scheduler executes tasks on an array of workers according to the specified dependencies. The user interface makes it easy to visualise pipelines running in production, monitor progress, and troubleshoot issues when needed.



DAGs

Show entries Search:

		DAG	Schedule	Owner	Recent Tasks	Last Run	DAG Runs	Links
	On	clm_flat_metadata	@once	airflow		2017-05-17 07:00		
	On	clm_flat_organisation	@once	airflow		2017-05-17 07:00		
	On	clm_import_metadata	None	airflow		2017-05-17 07:36		
	On	clm_pre_process_images	None	airflow		2017-05-17 07:36		
	On	clm_reorganise	None	airflow		2017-05-17 07:31		
	On	edsd_flat_metadata	@once	airflow		2017-05-17 07:00		
	On	edsd_flat_organisation	@once	airflow		2017-05-17 07:00		
	On	edsd_import_metadata	None	airflow		2017-05-18 09:54		
	On	edsd_pre_process_images	None	airflow		2017-05-17 09:24		
	On	edsd_reorganise	None	airflow		2017-05-17 07:31		
	On	mri_notify_failed_processing	None	airflow				
	On	mri_notify_skipped_processing	None	airflow		2017-05-18 00:59		
	On	mri_notify_successful_processing	None	airflow		2017-05-18 05:31		
	On	mri_self_checks	@hourly	airflow				

Showing 1 to 14 of 14 entries

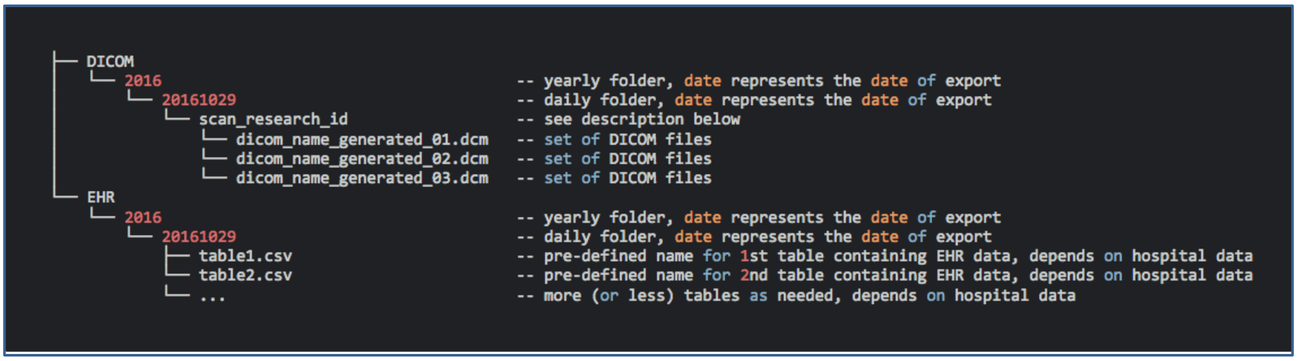
Previous 1 Next

Hide Paused DAGs

Figure 14: Apache Airflow Dashboard

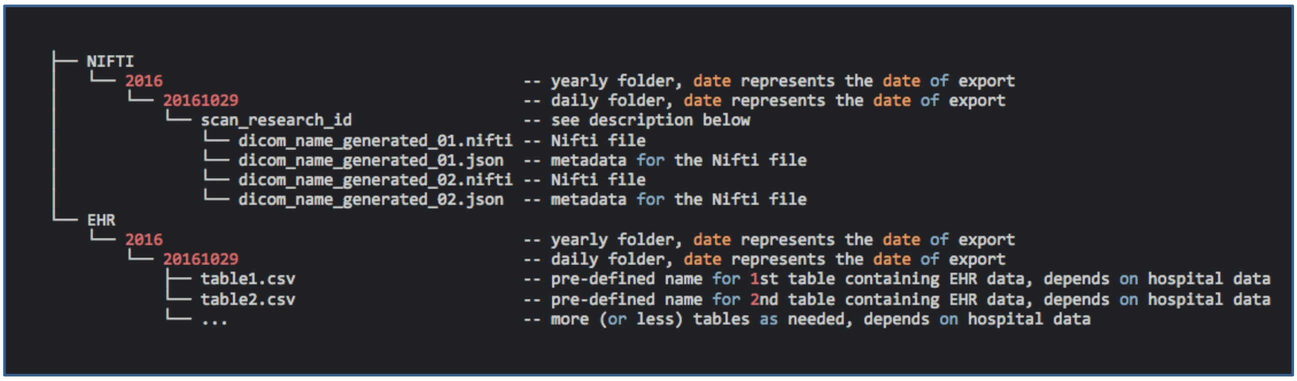
The Data Factory sub-system provides the following extraction, transformation and load functionality:

1) Pulling de-identified data out of the files stored in De-identified Data Version Control Storage



```
├── DICOM
│   └── 2016                                    -- yearly folder, date represents the date of export
│       └── 20161029                            -- daily folder, date represents the date of export
│           └── scan_research_id                -- see description below
│               └── dicom_name_generated_01.dcm -- set of DICOM files
│               └── dicom_name_generated_02.dcm -- set of DICOM files
│               └── dicom_name_generated_03.dcm -- set of DICOM files
└── EHR
    └── 2016                                    -- yearly folder, date represents the date of export
        └── 20161029                            -- daily folder, date represents the date of export
            ├── table1.csv                      -- pre-defined name for 1st table containing EHR data, depends on hospital data
            ├── table2.csv                      -- pre-defined name for 2nd table containing EHR data, depends on hospital data
            └── ...                             -- more (or less) tables as needed, depends on hospital data
```

Figure 15: De-identified DICOM and EHR Data



```
├── NIFTI
│   └── 2016                                     -- yearly folder, date represents the date of export
│       └── 20161029                             -- daily folder, date represents the date of export
│           └── scan_research_id                 -- see description below
│               └── dicom_name_generated_01.nifti -- Nifti file
│               └── dicom_name_generated_01.json  -- metadata for the Nifti file
│               └── dicom_name_generated_02.nifti -- Nifti file
│               └── dicom_name_generated_02.json  -- metadata for the Nifti file
└── EHR
    └── 2016                                     -- yearly folder, date represents the date of export
        └── 20161029                             -- daily folder, date represents the date of export
            ├── table1.csv                       -- pre-defined name for 1st table containing EHR data, depends on hospital data
            ├── table2.csv                       -- pre-defined name for 2nd table containing EHR data, depends on hospital data
            └── ...                              -- more (or less) tables as needed, depends on hospital data
```

Figure 16: De-identified NIfTI and EHR Data

2) Processing de-identified data to extract a patient's raw health-related features:
   a) Brain morphometric features (grey matter volume, shape and dimensions)
   b) Brain scan metadata
   c) Data from EHR files (demographic, biomarkers, neuropsychological assessments, diagnoses)
3) Harmonising data types from different source datasets into a common data element (CDE) model
4) Transformation of the extracted feature data and its permanent storage into the CDE Database
5) Placing feature data into files accessible by Features Data Store sub-system components

In addition to the components for extracting personal health features, the Data Factory sub-system contains a set of quality assurance components:

- **Quality Check** for a computational check of the quality of processed and extracted data
- **Imaging Plugin** to track all data changes during brain scan data processing and extraction
- **Data Tracking** to track all data changes except during brain scan data processing and extraction
- **Data Catalogue** to store data provenance/data version information

Reorganisation Pipeline

The Reorganisation pipeline is a component conditional to reorganise datasets pulled from the De-identified data version control storage to prepare them to enter the workflows for processing and extracting brain scan metadata, brain scan pre-processing and brain morphometric feature extraction and EHR data extraction.

The configuration of this pipeline needs to be tailored to every new hospital and research data set. The structure of the brain scan files (DICOM or NIfTI), including the metadata in their headers, depends on the non-standardised procedures specific for each hospital. The structure and the content of EHR files also need to be inspected, and configuration of the pipeline tailored accordingly.

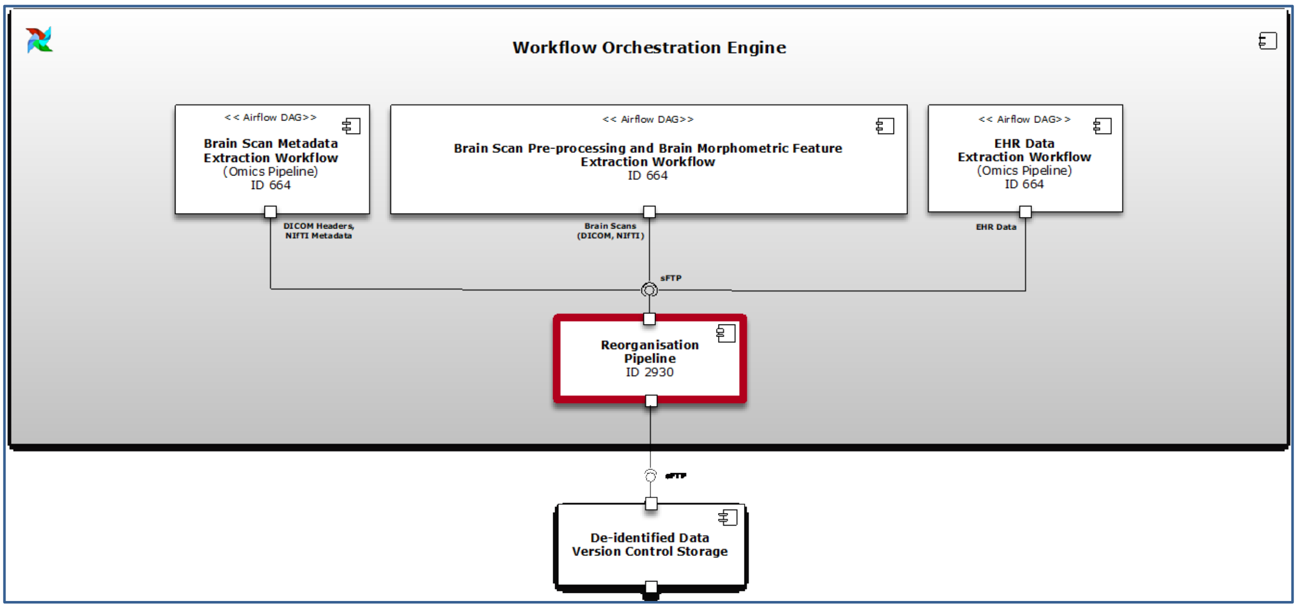

Workflow Orchestration Engine
<< Airflow DAG>>
Brain Scan Metadata Extraction Workflow
(Omics Pipeline)
ID 664
<< Airflow DAG>>
Brain Scan Pre-processing and Brain Morphometric Feature Extraction Workflow
ID 664
<< Airflow DAG>>
EHR Data Extraction Workflow
(Omics Pipeline)
ID 664
DICOM Headers, NIfTI Metadata
Brain Scans (DICOM, NIfTI)
EHR Data
sFTP
Reorganisation Pipeline
ID 2930
sFTP
De-identified Data Version Control Storage

Figure 17: Reorganisation Pipeline

Brain Scan Pre-processing and Brain Morphometric Feature Extraction Pipeline

Software systems are essential in all stages of neuroimaging, allowing scientists to control highly sophisticated imaging instruments and to make sense of the vast amounts of generated complex data. For magnetic resonance imaging (MRI), software systems are used to design and implement signal-capturing protocols in imaging instruments, reconstruct the resulting signals into a three-dimensional representation of the brain, correct for and suppress noise, statistically analyse the data, and visualise the results. Collected neuroimaging data can then be stored, queried, retrieved and shared using PACS, XNAT, CBRAIN, LORIS or any other system. Neuro-anatomical data can be extracted from neuroimages, compared and analysed using other specialised software systems, such as SPM and FreeSurfer.

After capturing and de-identifying neuroimaging DICOM data from PACS systems, the MIP's Data Factory sub-system extracts neuroanatomical data from captured brain magnetic resonance images, permanently stores that data into the Feature Data Store sub-system where it is made available for data mining and analysis together with the rest of biomedical and other health-related information.

The flow of data between Brain Scan Pre-processing and Brain Feature Extraction pipeline components is as follows:

1) **A visual quality check of the neuroimages performed by a neuroradiologist.**

   Pre-processing of magnetic resonance (MR) images strongly depends on the quality of input data. Multi-centre studies and data-sharing projects need to take into account varying image properties due to different scanners, sequences and protocols

   Image format requirements:

   - Full brain scans
   - Provided either in DICOM or NIFTI format
   - High-resolution (max. 1.5 mm) T1-weighted sagittal images.
   - If the dataset contains other types of images (that is not meeting the above description, e.g. fMRI data, T2 images, etc.), a list of protocol names used and their compatibility status regarding the above criterion has to be provided
   - Images must contain at least 40 slices

2) **The DICOM to NIfTI Converter converts brain scan data captured in DICOM format to NIfTI data format**

3) **The Neuromorphometric Processing component (SPM12) uses NIfTI data for computational neuro-anatomical data extraction using voxel-based statistical parametric mapping of brain image data sequences:**

   a) Each T1-weighted image is normalised to MNI (Montreal Neurological Institute) space using non-linear image registration SPM12 Shoot toolbox

   b) The individual images are segmented into three different brain tissue classes (grey matter, white matter and CSF)

   c) Each grey matter voxel is labelled based on Neuromorphometrics atlas (constructed by manual segmentation for a group of subjects) and the transformation matrix obtained in the previous step. Maximum probability tissue labels were derived from the "MICCAI 2012 Grand Challenge and Workshop on Multi-Atlas Labelling". These data were released under the Creative Commons Attribution-Non-Commercial (CC BY-NC. The MRI scans originate from the OASIS project, and the labelled data were provided by Neuromorphometrics, Inc. under an academic subscription

4) **The Voxel-Based Quantification (VBQ) component, through its sensitivity to tissue microstructure, provides absolute measures for neuroimaging biomarkers for myelination, water and iron levels comparable across imaging sites and in time**

5) **The I2B2 Import component stores extracted brain morphometric features in I2B2 Capture Database, alongside the brain scan metadata and patient EHR data**

The Quality Check component evaluates essential image parameters, such as signal-to-noise ratio, inhomogeneity and image resolution. It evaluates images for problems during the processing steps. It allows comparing quality measures across different scans and sequences.

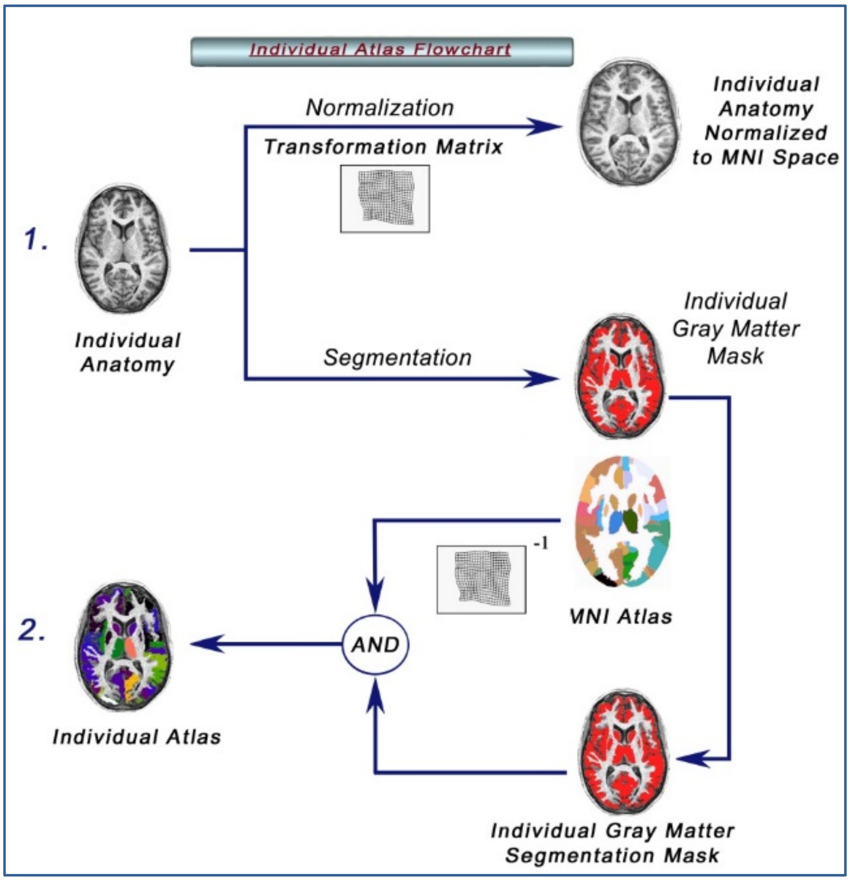
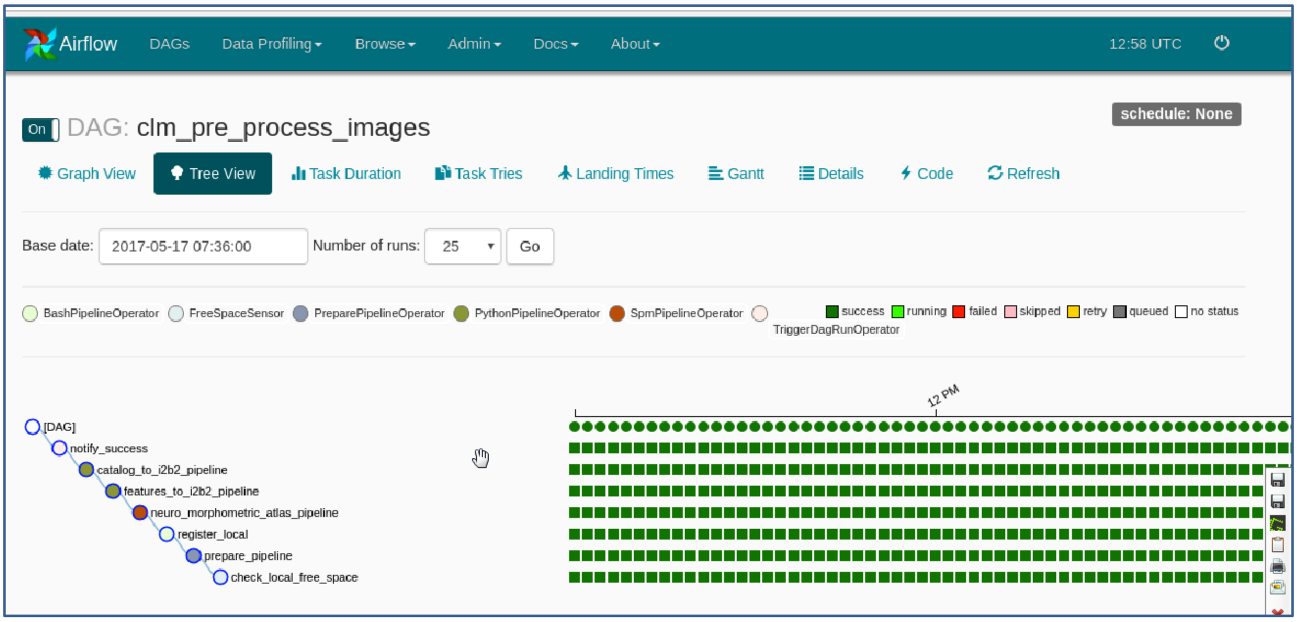

Individual Atlas Flowchart
Normalization
Transformation Matrix
Individual Anatomy Normalized to MNI Space
1.
Individual Anatomy
Segmentation
Individual Gray Matter Mask
-1
MNI Atlas
2.
AND
Individual Atlas
Individual Gray Matter Segmentation Mask

Figure 18: Neuromorphometric Processing



DAG: clm_pre_process_images

Base date: 2017-05-17 07:36:00 Number of runs: 25

[DAG]
notify_success
catalog_to_i2b2_pipeline
features_to_i2b2_pipeline
neuro_morphometric_atlas_pipeline
register_local
prepare_pipeline
check_local_free_space

Figure 19: Apache Airflow Image Processing Pipeline Status

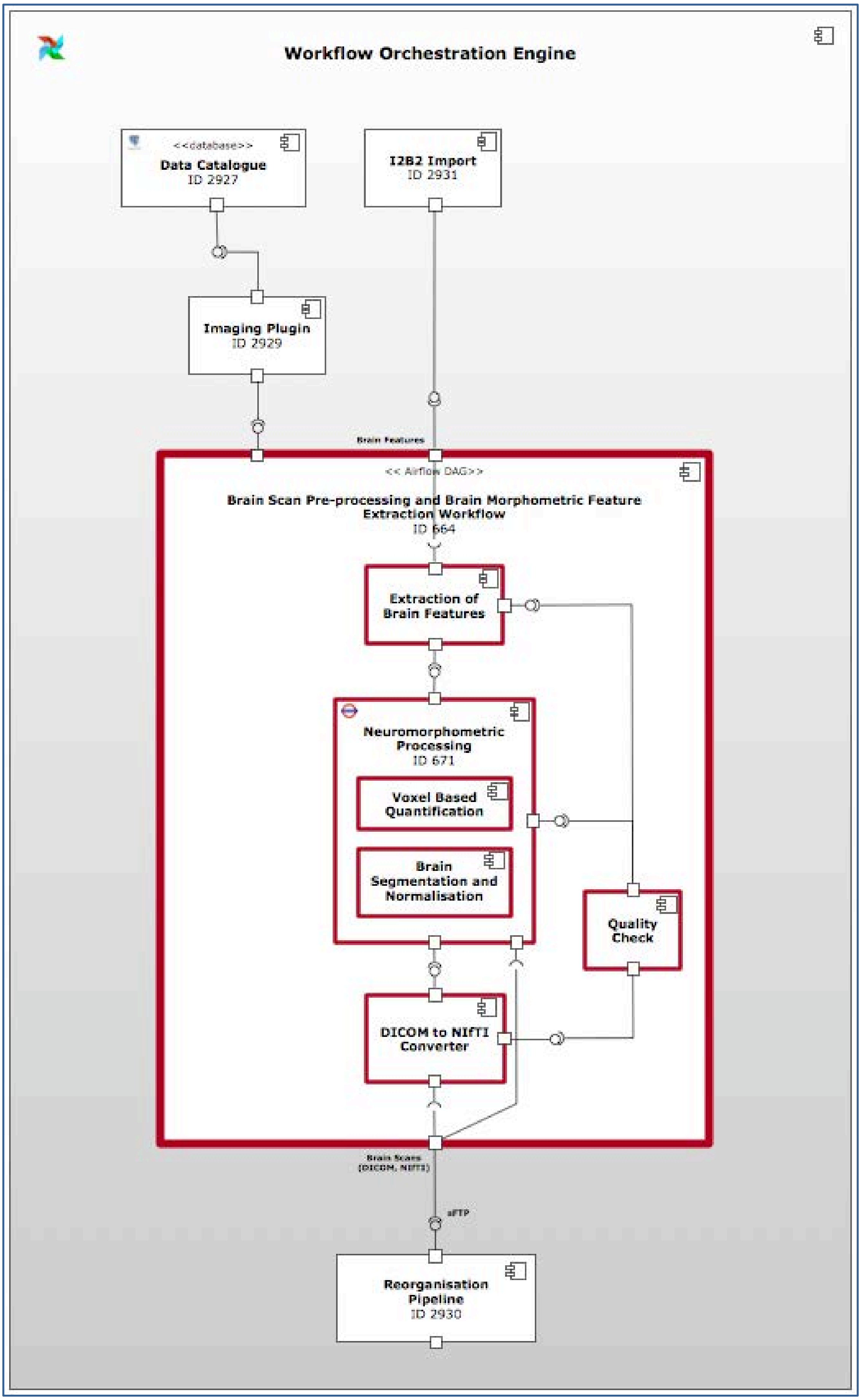

Workflow Orchestration Engine
<<database>>
Data Catalogue
ID 2927
I2B2 Import
ID 2931
Imaging Plugin
ID 2929
Brain Features
<< Airflow DAG>>
Brain Scan Pre-processing and Brain Morphometric Feature Extraction Workflow
ID 664
Extraction of Brain Features
Neuromorphometric Processing
ID 671
Voxel Based Quantification
Brain Segmentation and Normalisation
Quality Check
DICOM to NIfTI Converter
Brain Scans (DICOM, NIfTI)
sFTP
Reorganisation Pipeline
ID 2930

Figure 20: Brain Scan Pre-processing and Brain Feature Extraction Workflow

The Brain Scan Pre-processing and Brain Morphometric Feature Extraction pipeline contains components for processing T1-weighted brain image data sequences and extracting morphometric brain features – grey matter volume and shape – using voxel-based morphometry (VBM). VBM provides insight into macroscopic volume changes that may highlight differences between groups, be associated with pathology or be indicative of plasticity.

For neuromorphometric processing, the MIP uses SPM12 software running within the MATLAB software environment. For image pre-processing and morphometric feature extraction, SPM requires input data in a standard format used by neuromorphometric tools for computation and feature extraction: the NIfTI format.

The T1-weighted images are automatically segmented into 114 anatomical structures using the Neuromorphometrics atlas.

In addition to voxel-based neuromorphometric processing of T1-weighted images for classification of tissue types and measuring of macroscopic anatomical shape, the MIP uses a voxel-based quantification (VBQ) toolbox as a plugin for SPM12 that can analyse high-resolution quantitative imaging and can provide neuroimaging biomarkers for myelination, water and iron levels that are absolute measures comparable across imaging sites and in time.

Single NIfTI volumes of the brain are first partitioned into three classes: grey matter, white matter and background. This procedure also incorporates an approximate image alignment step and a correction for image intensity non-uniformities. This procedure uses the SPM12 Segment5 tool.

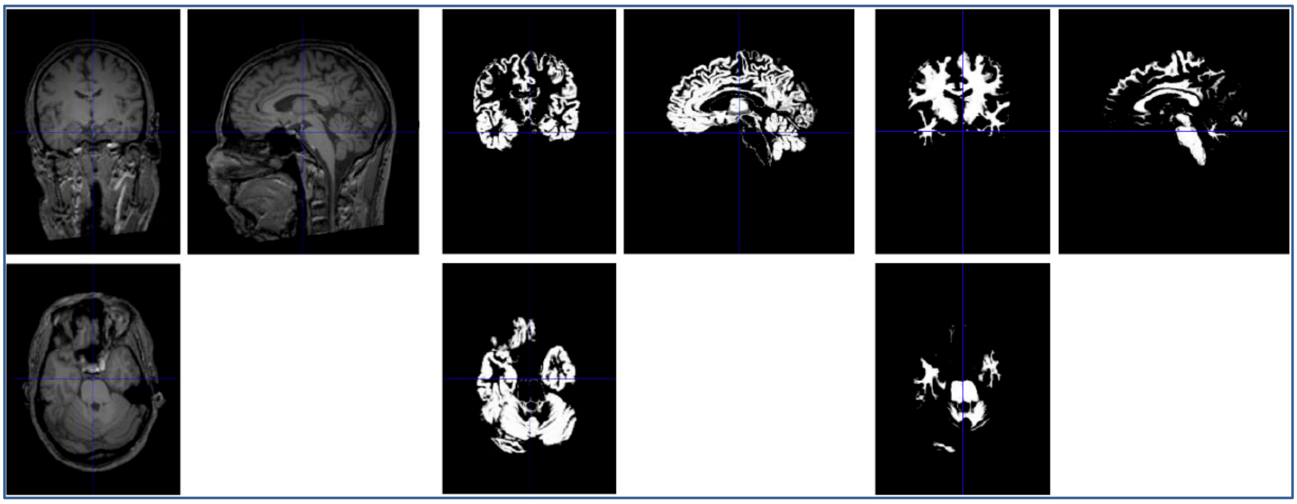


Figure 21: Original T1-weighted MRI scan (left), along with automatically extracted grey (middle) and white matter (right) tissue maps. The tissue maps encode the probability of each tissue type calculated using the given model and data

Tissue atlases, pre-computed from training data are then spatially registered with the extracted grey and white matter maps, using the Shoot5 tool from SPM12. The warps estimated from this registration step are then used to project other pre-computed image data into alignment with the original scans (and their grey and white matter maps).

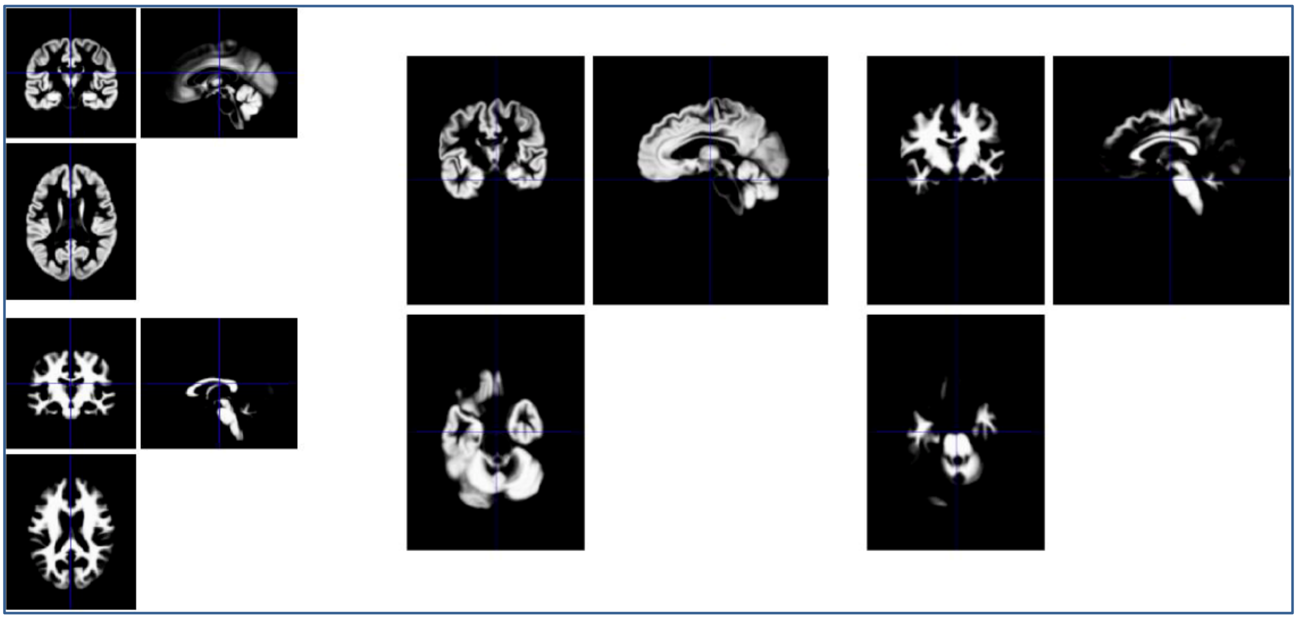


Figure 22: Grey and white matter from the original tissue atlases (left) along with registered versions (middle and right)

The rules of probability are then used to combine the various images to give a probabilistic label map for each brain structure. These probabilities are summed for each structure, to provide probabilistic volume estimates. These estimates are saved in the MIP platform as brain morphometric features.

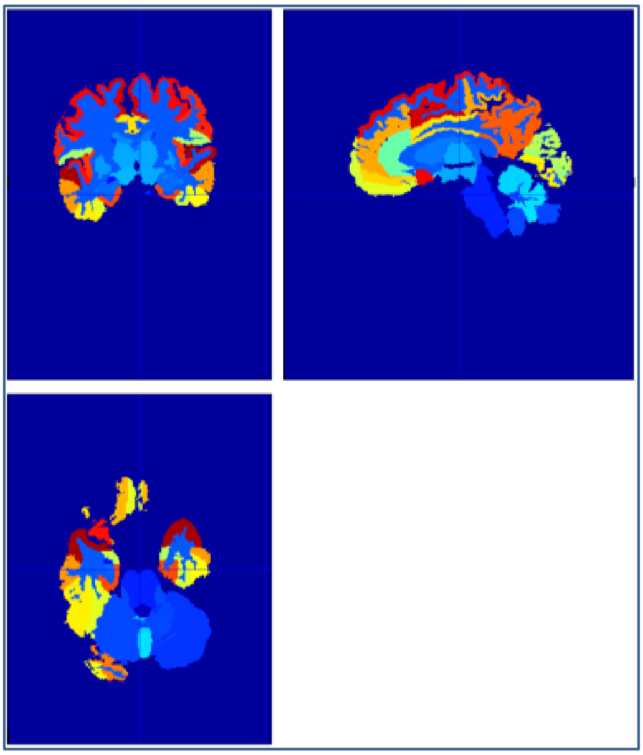


Figure 23: Automatically labelled image, showing most probable macro anatomy structure labels

While Voxel-based morphometry classifies tissue types and measures anatomical shape (Brain Segmentation and Normalisation component), the Voxel-Based Quantification component provides complementary information through its sensitivity to tissue microstructure. The Multi-parameter Mapping (MPM) imaging protocol is used to provide whole-brain maps of relaxometry measures ($R_1 = 1/T_1$ and $R_2* = 1/T_2*$), magnetisation transfer saturation (MT) and effective proton density (PD*) with the isotropic resolution of 1mm or higher.

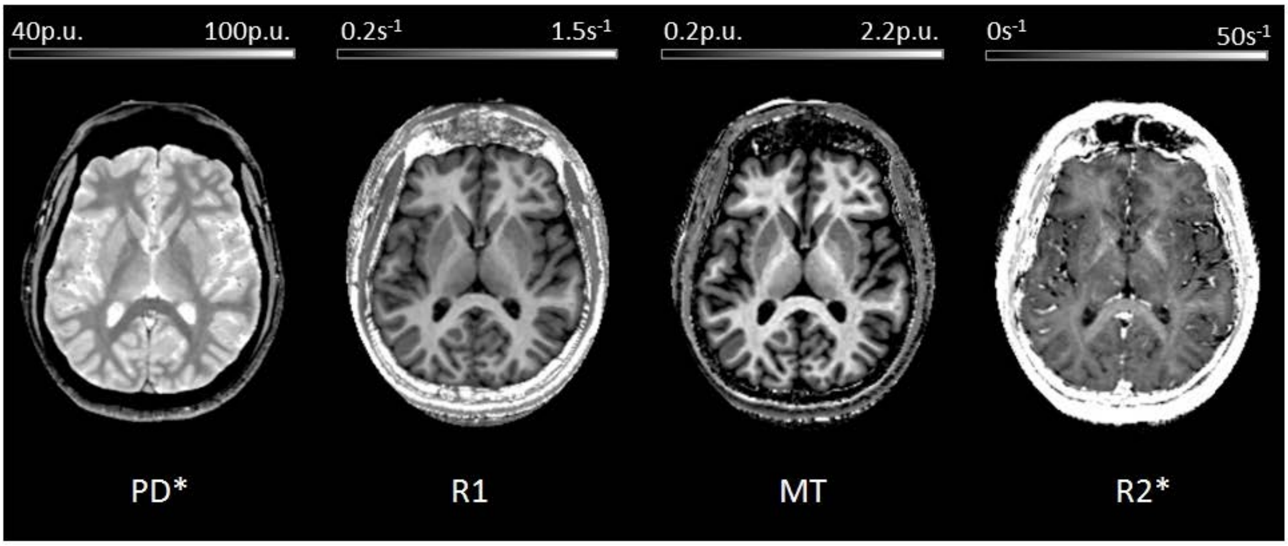


Figure 24: Multi Parameter Mapping high-resolution quantitative MRI acquisition protocol

MPM is a high-resolution quantitative imaging MRI protocol which, combined with VBQ data analysis, opens new windows for studying the microanatomy of the human brain *in vivo.* With T1-weighted images, the signal intensity is in arbitrary units and cannot be compared across sites or even scanning sessions. Quantitative imaging can provide absolute measures for neuroimaging biomarkers for myelination, water and iron levels comparable across imaging sites and in time.

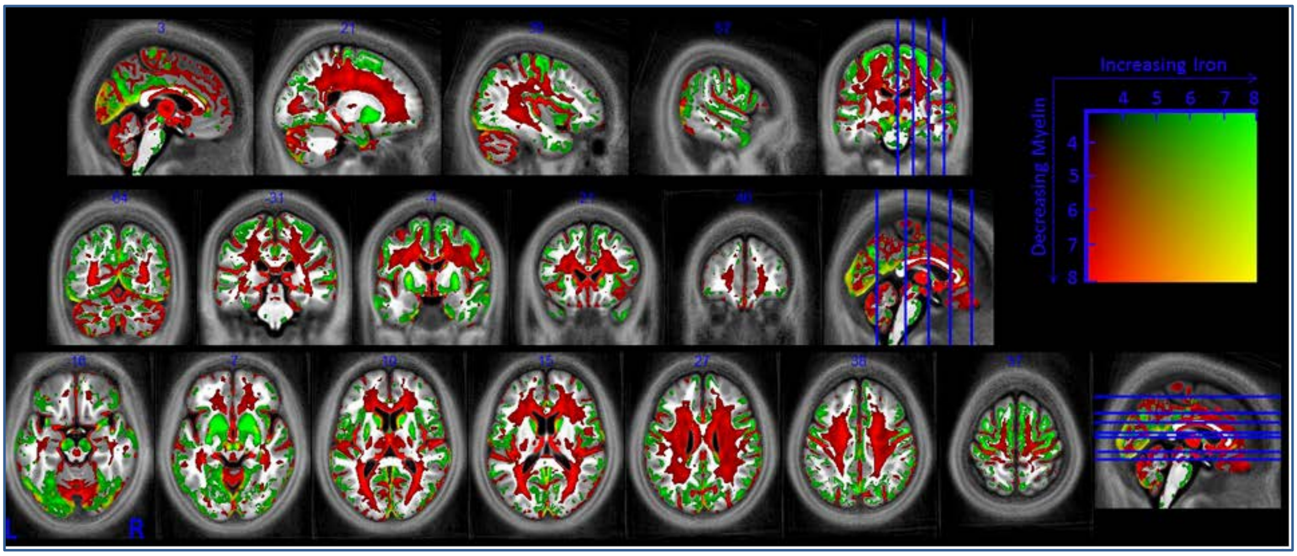


Figure 25: Voxel Based Quantification data analysis for studying microanatomy of the human brain *in vivo*

Brain Scan Metadata Extraction and EHR Data Extraction Pipelines

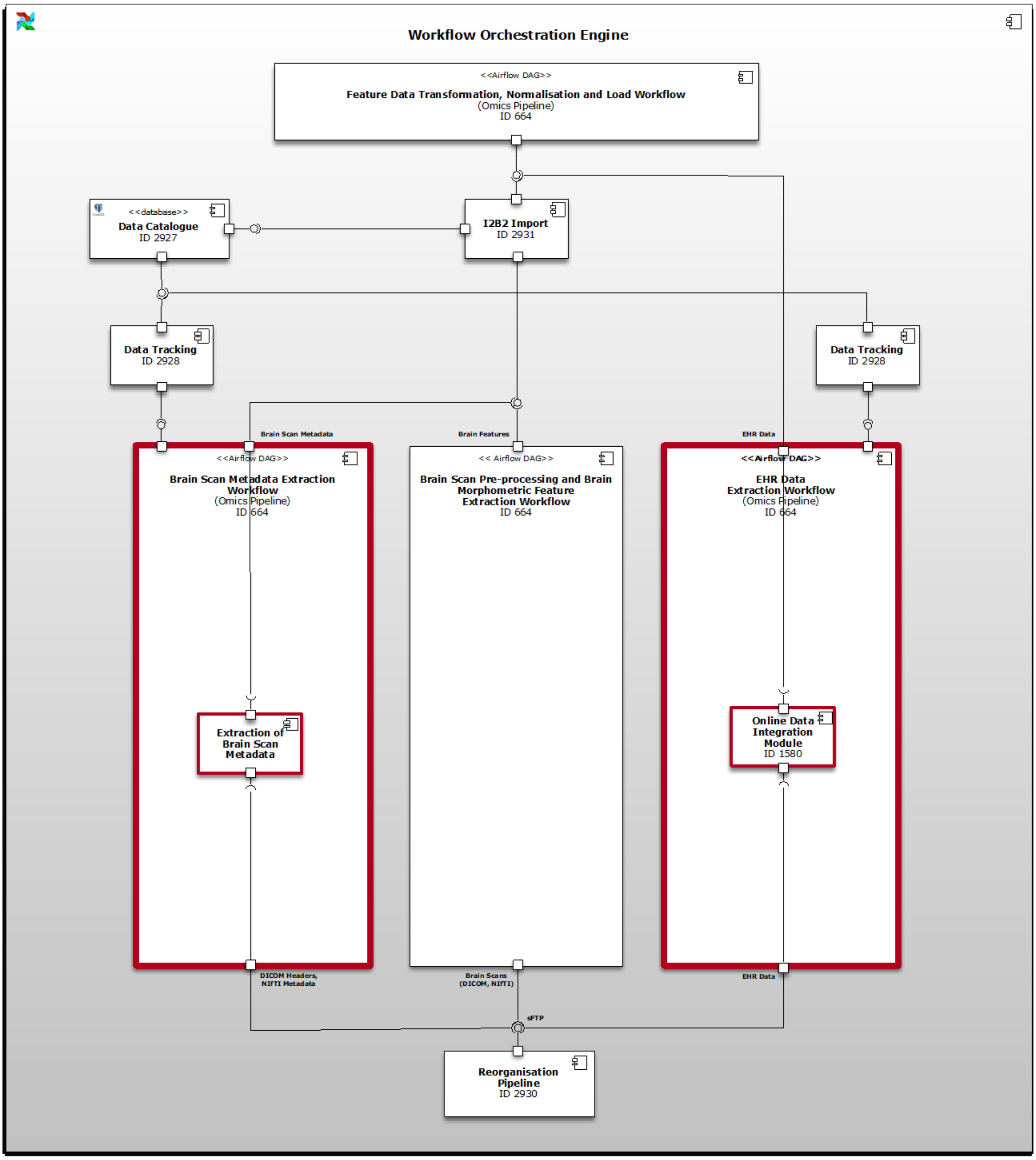

Workflow Orchestration Engine
<<Airflow DAG>>
Feature Data Transformation, Normalisation and Load Workflow
(Omics Pipeline)
ID 664
<<database>>
Data Catalogue
ID 2927
I2B2 Import
ID 2931
Data Tracking
ID 2928
Data Tracking
ID 2928
Brain Scan Metadata
Brain Features
EHR Data
<<Airflow DAG>>
Brain Scan Metadata Extraction Workflow
(Omics Pipeline)
ID 664
<< Airflow DAG>>
Brain Scan Pre-processing and Brain Morphometric Feature Extraction Workflow
ID 664
<<Airflow DAG>>
EHR Data Extraction Workflow
(Omics Pipeline)
ID 664
Extraction of Brain Scan Metadata
Online Data Integration Module
ID 1580
DICOM Headers, NIFTI Metadata
Brain Scans (DICOM, NIfTI)
EHR Data
sFTP
Reorganisation Pipeline
ID 2930

Figure 26: Brain Scan Metadata and EHR Data extraction pipelines

A patient's brain scan metadata and EHR data are extracted from the corresponding de-identified files and stored in I2B2 Capture Database alongside extracted brain morphometric features. Data provenance is stored in Data Catalogue.

Feature Data Transformation, Normalisation and Load Pipeline

This pipeline contains the following components:

- **Data Capture Database** – for storing patient health features extracted from brain scans and EHR files
- **Data Mapping and Transformation Specification** – data mapping rules – the results of harmonising data types from different source datasets into a common data element (CDE) model

- **Online Data Integration Module** – for transformation of the extracted patient feature data into the common data elements format, according to the Data Mapping and Transformation Specification rules. Also for exporting CDE Database to CSV file for storing the harmonised data into the local data store mirror (Features Table) in Features Data Store sub-system
- **Common Data Elements Database** – for permanently storing the transformed patient feature data into a normalised I2B2 schema

Data Capture Database

De-identified data, extracted from patient electronic health records and brain scans, is stored in the original data format in the Data Capture Database, implemented using I2B2 schema managed by PostgreSQL database management system.

The I2B2 schema allows for an optional direct update of Data Capture Database with data from a large number of I2B2-compliant anonymised patient cohort datasets. I2B2 is widely used for implementing clinical data warehouses as well as research data warehouses. Over the years, it became a de facto standard for bridging the gap between clinical informatics and bioinformatics, providing large datasets for clinical, biomedical and pharmaceutical research.

In cases when research datasets are stored in different formats, such as ADNI or BIDS files, they are initially saved in the Data Factory sub-system's version controlled storage before the data is extracted using the extraction pipelines and then finally stored in the Capture Database.

Data Mapping and Transformation Specification

The MIP Data Governance and Data Specification (DGDS) team receive information from hospitals about new data elements that shall be captured from patient EHR and brain scan datasets. In collaboration with hospital clinicians and data managers, the MIP DGDS team analyses new data types and harmonises them into a common data elements model. Data Mapping and Transformation Specification is updated with new harmonisation rules. This artefact is used for transformation of original data extracted from hospitals into the common data element format using the Online Data Integration Module.

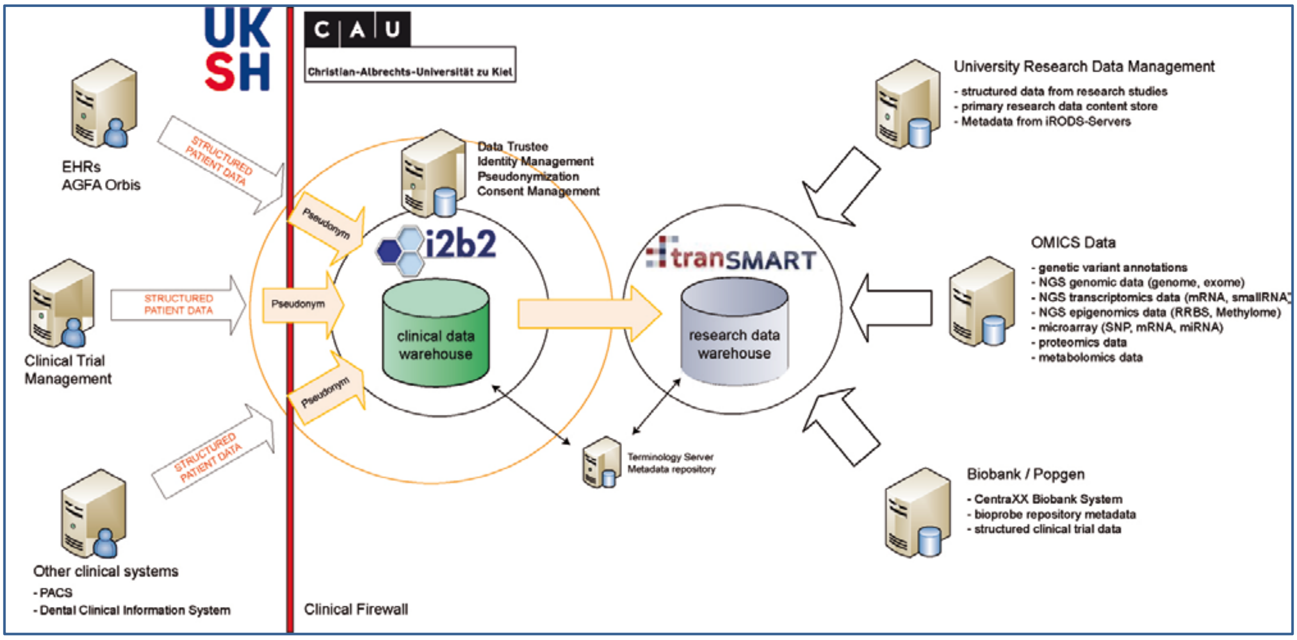
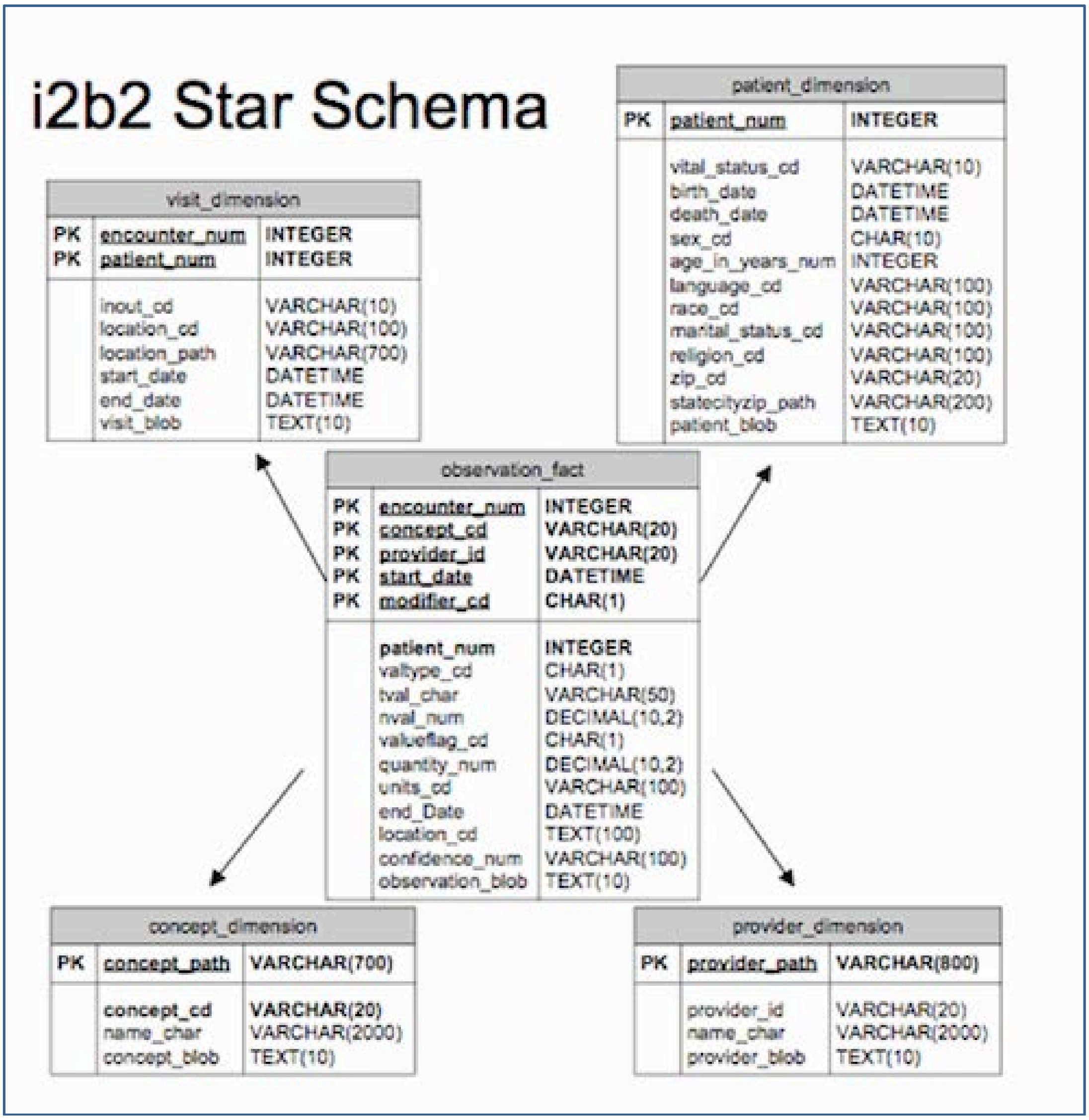

UK SH
C A U
Christian-Albrechts-Universität zu Kiel
EHRs
AGFA Orbis
STRUCTURED PATIENT DATA
Clinical Trial Management
STRUCTURED PATIENT DATA
STRUCTURED PATIENT DATA
Other clinical systems
- PACS
- Dental Clinical Information System
Clinical Firewall
Pseudonym
Pseudonym
Pseudonym
Data Trustee
Identity Management
Pseudonymization
Consent Management
i2b2
clinical data warehouse
tranSMART
research data warehouse
Terminology Server
Metadata repository
University Research Data Management
- structured data from research studies
- primary research data content store
- Metadata from iRODS-Servers
OMICS Data
- genetic variant annotations
- NGS genomic data (genome, exome)
- NGS transcriptomics data (mRNA, smallRNA)
- NGS epigenomics data (RRBS, Methylome)
- microarray (SNP, mRNA, miRNA)
- proteomics data
- metabolomics data
Biobank / Popgen
- CentraXX Biobank System
- bioprobe repository metadata
- structured clinical trial data

Figure 27: I2B2 tranSMART Foundation's research data warehouse for clinical, biomedical and pharmaceutical research



i2b2 Star Schema

patient_dimension		
PK	patient_num	INTEGER
	vital_status_cd	VARCHAR(10)
	birth_date	DATETIME
	death_date	DATETIME
	sex_cd	CHAR(10)
	age_in_years_num	INTEGER
	language_cd	VARCHAR(100)
	race_cd	VARCHAR(100)
	marital_status_cd	VARCHAR(100)
	religion_cd	VARCHAR(100)
	zip_cd	VARCHAR(20)
	statecityzip_path	VARCHAR(200)
	patient_blob	TEXT(10)

visit_dimension		
PK	encounter_num	INTEGER
PK	patient_num	INTEGER
	inout_cd	VARCHAR(10)
	location_cd	VARCHAR(100)
	location_path	VARCHAR(700)
	start_date	DATETIME
	end_date	DATETIME
	visit_blob	TEXT(10)

observation_fact		
PK	encounter_num	INTEGER
PK	concept_cd	VARCHAR(20)
PK	provider_id	VARCHAR(20)
PK	start_date	DATETIME
PK	modifier_cd	CHAR(1)
	patient_num	INTEGER
	valtype_cd	CHAR(1)
	tval_char	VARCHAR(50)
	nval_num	DECIMAL(10,2)
	valueflag_cd	CHAR(1)
	quantity_num	DECIMAL(10,2)
	units_cd	VARCHAR(100)
	end_Date	DATETIME
	location_cd	TEXT(100)
	confidence_num	VARCHAR(100)
	observation_blob	TEXT(10)

concept_dimension		
PK	concept_path	VARCHAR(700)
	concept_cd	VARCHAR(20)
	name_char	VARCHAR(2000)
	concept_blob	TEXT(10)

provider_dimension		
PK	provider_path	VARCHAR(800)
	provider_id	VARCHAR(20)
	name_char	VARCHAR(2000)
	provider_blob	TEXT(10)

Figure 28: I2B2 Schema

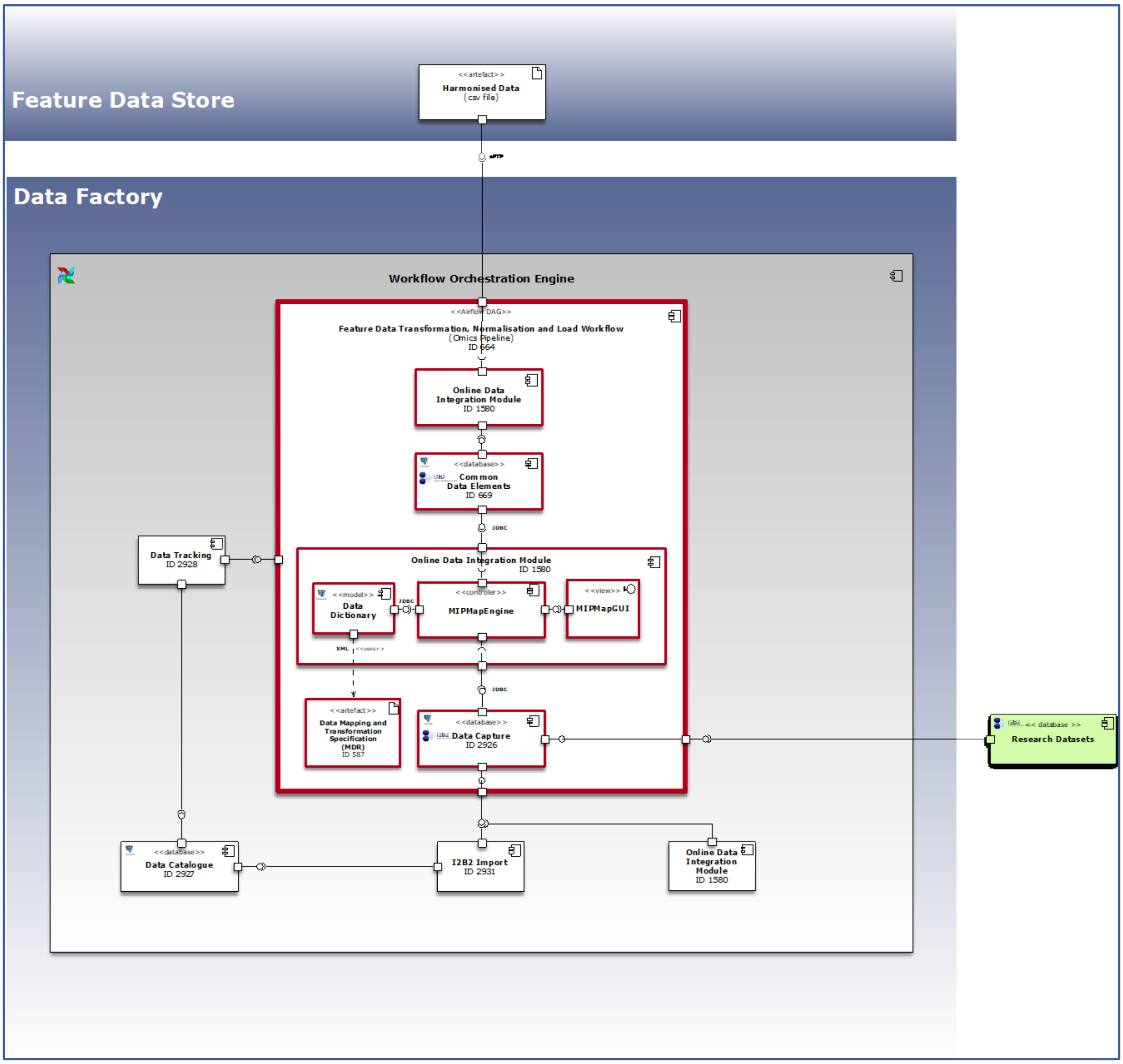
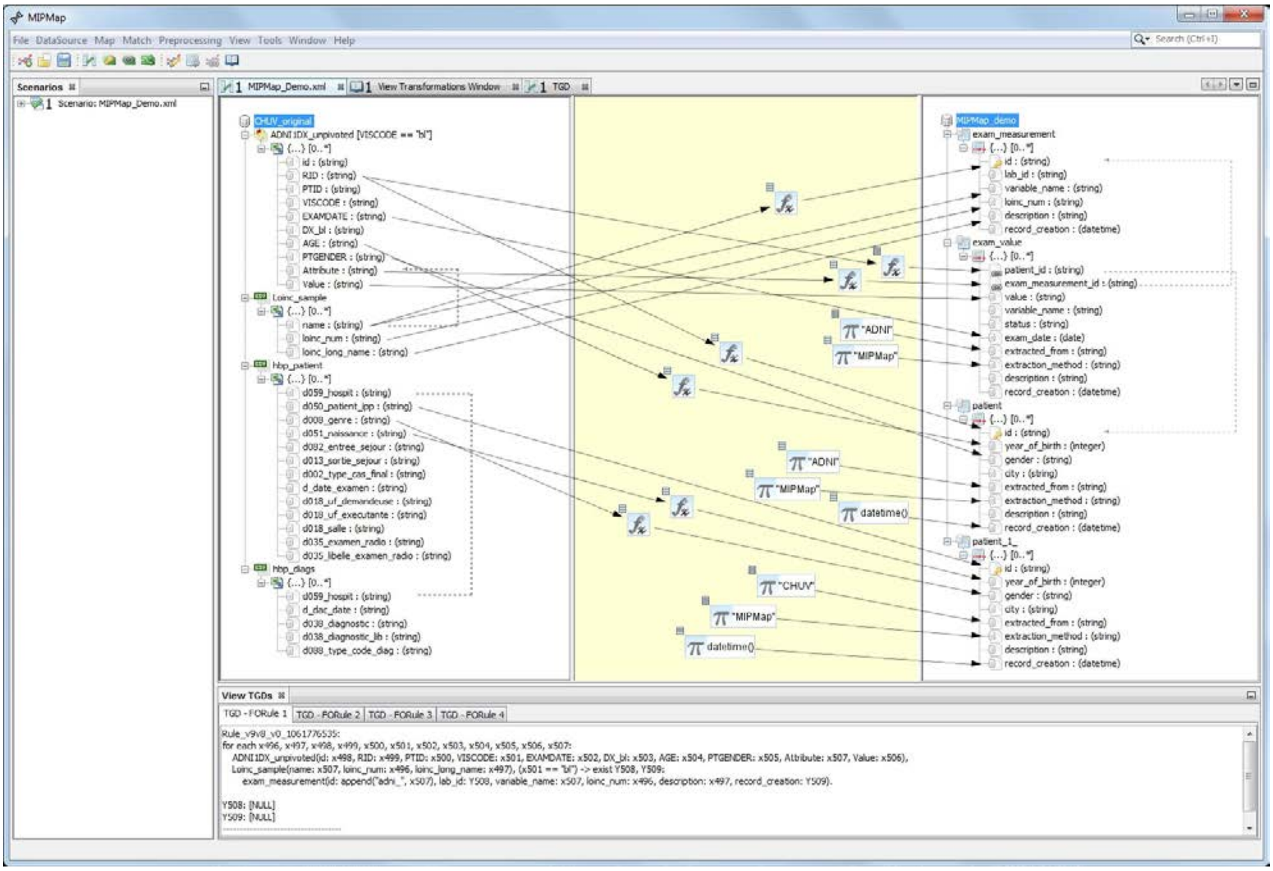


Figure 29: Feature Data Transformation, Normalisation and Load Pipeline

Online Data Integration Module for Data Transformation and Load to CDE Database

The Online Data Integration Module component is used for extracted data transformation, and loading into the normalised I2B2-compliant Common Data Element Database, managed by PostgreSQL database management system. This component is also used to export harmonised data from CDE Database to CSV files, out of which the Feature Table in the Feature Data Store sub-system is populated. The Online Data Integration Module is implemented using an open source ++Spicy data exchange tool. The adaptation of this application for the MIP is called MIPMap. This tool, which has been developed in Java using the NetBeans platform, applies Data Mapping and Transformation Specification rules for transformation of data stored in I2B2 Capture Database to the normalised I2B2 Common Data Elements Database.

MIPMap provides a graphical user interface where a hospital data manager or a MIP DGDS data manager can create mapping correspondences between source data elements and targets by drawing lines between them. This forms a mapping scenario that is stored in XML format. The mapping process is performed once for every hospital.



```
Rule_v9v8_v0_1061776535:
for each x496, x497, x498, x499, x500, x501, x502, x503, x504, x505, x506, x507:
   ADNI1DX_unpivoted(id: x498, RID: x499, PTID: x500, VISCODE: x501, EXAMDATE: x502, DX_bl: x503, AGE: x504, PTGENDER: x505, Attribute: x507, Value: x506),
   Loinc_sample(name: x507, loinc_num: x496, loinc_long_name: x497), (x501 == "bl") -> exist Y508, Y509:
      exam_measurement(id: append("adni_", x507), lab_id: Y508, variable_name: x507, loinc_num: x496, description: x497, record_creation: Y509).

Y508: [NULL]
Y509: [NULL]
```

Figure 30: MIPMap user interface

Having created a mapping scenario, the MIPMap Engine generates an optimised SQL script that translates the data from the source (CSV file or a database schema) to the target database schema and then updates the target database.

Common Data Elements Database (delivered by CHUV team)

After the source data types have been mapped to the destination common data elements schema using the Data Mapping and Transformation Specification, the Online Data Integration Module loads the data from the Capture Database to the Common Data Elements Database.

An I2B2-compliant Common Data Elements (CDE) database schema is incorporated on top of the PostgreSQL database management system for permanently storing harmonised patient data from different hospitals and research datasets.

One of the key added-value characteristics of the MIP is the harmonisation of data elements from diverse source systems – EHR systems from different hospitals, imaging and PACS systems and research datasets. The harmonised data model is implemented as an I2B2-compliant database schema, which allows for a prospective easy integration with a large research datasets compliant with I2B2.

Online Data Integration Module for Transformation of CDE Database to Harmonised Data CSV File

Harmonised data from the CDE Database is transformed using the Online Data Integration Module component into a Harmonised Data CSV File in the Feature Data Store sub-system. The MIPMap Engine executes a pivoting script, for pivoting the variables and their values stored in the dimensional I2B2 (data mart) schema of CDE Database into a flat comma-separated value representation. The Harmonised Data CSV File is processed by the Query Engine and stored in the Feature Table to be available to the components of the Knowledge Extraction sub-system for data mining, statistical analysis and predictive machine learning.

3.1.3 Feature Data Store Subsystem

The Feature Data Store Sub-system contains components for mirroring harmonised patient data in the form appropriate for querying and using by machine learning algorithms. The components of this subsystem operate on and store the data belonging to one and only one hospital. The data is made available both for the local knowledge extraction MIP subsystem and to the remote, federated knowledge extraction MIP sub-system.

The components of the Feature Data Store sub-system are as follows:

- **Harmonised Data CSV file** – for mirroring harmonised CDE data exported from CDE database
- **Query Engine** – hospital DB back end, executing queries on extracted patient health sensitive data
- **Features Database** – hospital local data store mirror, data ready for querying and machine learning
- **PostgresRAW-UI** – user interface for Query Engine administration, including CSV files monitor

Harmonised Data CSV File

Using the Online Data Integration Module component, harmonised de-identified health-related patient data is exported from the CDE Database in the Data Factory sub-system into the CSV files accessible from the Feature Data Store sub-system components. The Query Engine component queries data stored in these files. The Query Engine also makes the data available for fetching by data mining and machine learning algorithms by storing it in the Hospital Dataset table of the Features Database.

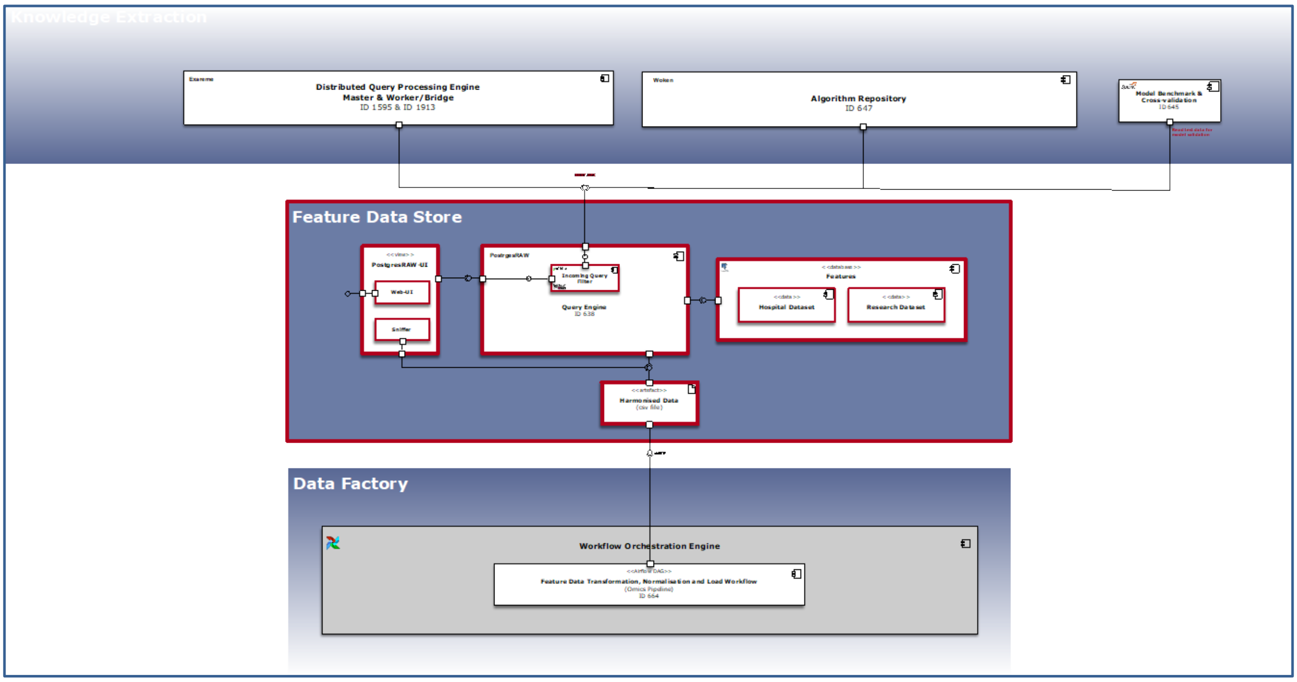


Figure 31: Feature Data Store Sub-system

Query Engine

The main purpose of the Query Engine component is to provide querying of the harmonised patient data stored in CSV files. The MIP Query Engine component is a database management system named PostgresRAW, based on PostgreSQL.

The input to the Query engine is data stored in CSV files. The output of the Query Engine is provided in JSON file format using REST services API or regular PostgreSQL connections.

Features Database

The Flat Hospital Dataset Table of the Features Database is updated with the data queried directly from the files. There it is made available for further querying by the Distributed Query Processing Engine or fetching by machine learning algorithms from the Algorithm Factory, both in the Knowledge Extraction sub-system.

The querying and fetching of data from the Feature Database is performed locally. For the privacy reasons, de-identified patient data is not allowed to be copied outside the hospital's MIP execution environment. The necessary computation is distributed throughout the hospital environments and only the results are fetched by the federation execution environment, either for visualisation or for further processing.

In addition to the Hospital Dataset flat table, the Features Database contains the Research Dataset flat table populated with the data captured from open research cohort datasets.

PostgresRAW-UI

PostgresRAW-UI automates detection and registration of raw files by providing a file monitor (Sniffer component). The folder containing the files with data that should update the Hospital Dataset table is provided as an argument when starting the database server.

3.1.4 Knowledge Extraction Subsystem

The components of the Knowledge Extraction sub-system are deployed both within the local hospital MIP execution environments and within the central MIP federation execution environment.

This MIP sub-system provides the functions for processing of the harmonised patient data, for local or distributed data mining and local or distributed execution of statistical inference and machine learning algorithms.

The two major complementary components of Knowledge Extraction sub-system are:

- **Algorithm Factory (Woken)** – orchestration of machine learning algorithm execution, including model benchmarking and cross-validation and storing of the trained models and their estimated predictive errors. Does not have out-of-the-box support for database query processing
- **Distributed Query Processing Engine (Exareme)** – query processing orchestration engine optimised for execution of distributed database queries extended with user-defined functions. Does not have out-of-the-box support for estimating trained machine learning model predictive errors

3.1.4.1 Algorithm Factory

Algorithm Orchestrator (Woken)

This component is a workflow orchestration platform, which runs statistical, data mining and machine learning algorithms encapsulated in Docker containers. Algorithms and their runtime are fetched from the Algorithm Repository, a Docker registry containing approved and compatible algorithms and their runtimes and libraries.

This component runs on top of the runtime environment containing Mesos and Chronos to control and execute the Docker containers over a cluster.

This component provides a web interface for on-demand execution of algorithms. It fetches the algorithms from the Algorithm Repository, monitors the execution of the algorithms also from the other execution environments in the cluster, collects the results formatted as a PFA document and returns a response to the web front end.

The Algorithm Orchestrator tracks data provenance information, runs model benchmarking and cross-validation of the models learned by the machine learning algorithms, using random K-Fold Sampling methods (Model Benchmark & Cross-validation), and stores PFA models in the Predictive Disease Model Repository.

Algorithm Repository

This component is a repository of Docker images that can be used by the Algorithm Orchestrator. It provides a workflow that allows contributors to provide new algorithms in a secured manner.

Algorithms, written in their native language (Python, MATLAB, R, Java, etc.), are encapsulated in a Docker container that provides them with the libraries and runtime environment necessary to execute this function. Currently, the MIP SGA1 platform supports Python-, Java- and R-based algorithms that are packaged in three Docker containers, respectively. The environment variables provided to the Docker container are used as algorithm parameters.

Algorithm Docker containers are autonomous:

- Connecting to the Features Database in the Features Data Store sub-system to retrieve feature data
- Processing data, taking into account Docker container environment variables
- Storing results into the Predictive Disease Model Repository

The Algorithm Registry database, implemented using PosgtreSQL database management system, is used to keep track of results created by the execution of an algorithm.

New algorithms can be easily integrated with the others by packaging them in the relevant Docker container. The supported algorithm results format is PFA, described in YAML or JSON configuration file. PFA enables vendor-neutral exchange and execution of complex predictive machine learning models. For visualisations, MIP SGA1 supports different formats, including Highcharts, Vis.js, PNG and SVG.

Machine learning algorithms planned for integrated by the end of SGA1 phase are:

Table 4: List of supported machine learning algorithms

Name	Methods	Federation/Local	PFA cross-validation
java-jsi-clus-fire	Clustering methods	Local	no
java-jsi-clus-fr	Clustering methods	Local	no
java-jsi-clus-pct-ts	Clustering methods	Local	no
java-jsi-clus-pct	Clustering methods	Local	yes
java-jsi-streams-modeltree	Tree-based methods	local	yes
java-jsi-streams-regressiontree	Tree-based methods	Local	yes
java-rapidminer-knn	Classification	Local	yes
java-rapidminer-naivebayes	Classification	Local	yes
python-anova	Classical inference	Local and Federation	yes
python-correlation-heatmap	Classical inference	Local and Federation	no
python-distributed-kmeans	Clustering	Local and Federation	yes
python-histograms	Descriptive	Local and Federation	no
python-jsi-hedwig	Tree-based	Local	no
python-jsi-hinmine	Tree-based	Local	no
python-knn	Classification	Local	yes
python-linear-regression	Predictive linear regression	Local and Federation	yes
python-longitudinal	Longitudinal analyses	Local	yes
python-sgd-regression	Gradient descent	Local and Federation	yes

Name	Methods	Federation/Local	PFA cross-validation
python-summary-statistics	Descriptive	Local and Federation	no
python-tsne	descriptive	Local	yes
r-3c	classification	Local	no
r-ggparci	Exploration	Local	no
r-heatmaply	Correlation	Local	no
r-linear-regression	Baysesian regression	Local and Federation	yes
Exareme k-means	Clustering	Federation	no
Exareme regression	Regression	Federation	no

Model Benchmark & Cross-validation

The Model benchmark and Cross-validation component is used to measure machine-learning models' accuracy. The results can guide the user to select the best-performing algorithm and fine-tune its parameters as well as to understand how well the model performs before it's used in production.

A model trained on training data needs to be validated. Its quality is measured by estimating its predictive error. Several techniques for assessing predictive errors exist, cross-validation being the most frequently used one. The predictive error is calculated by using the two disjoint datasets – training data set, to train the model, and test dataset to calculate the predictive error rate. The calculation of model predictive error rates is called validation.

Data used for both training and test datasets are stored in the Features Database, in the Features Data Store sub-system. The Model Benchmark & Cross-validation component performs data split using K-Fold cross-validation. This method of data sampling divides the complete dataset into K disjoint parts of roughly the same size. K different models are trained on K-1 parts each, while being tested on the remaining one part of the data. That is done on all K parts exactly once to ensure that every data row is used equally often for training and exactly once for testing. Resulting K test errors are then averaged to get the final error estimate of the model, which was built on the complete dataset.

The Algorithm Orchestrator stores the trained machine learning models and the results of cross-validations in the Predictive Disease Model Repository.

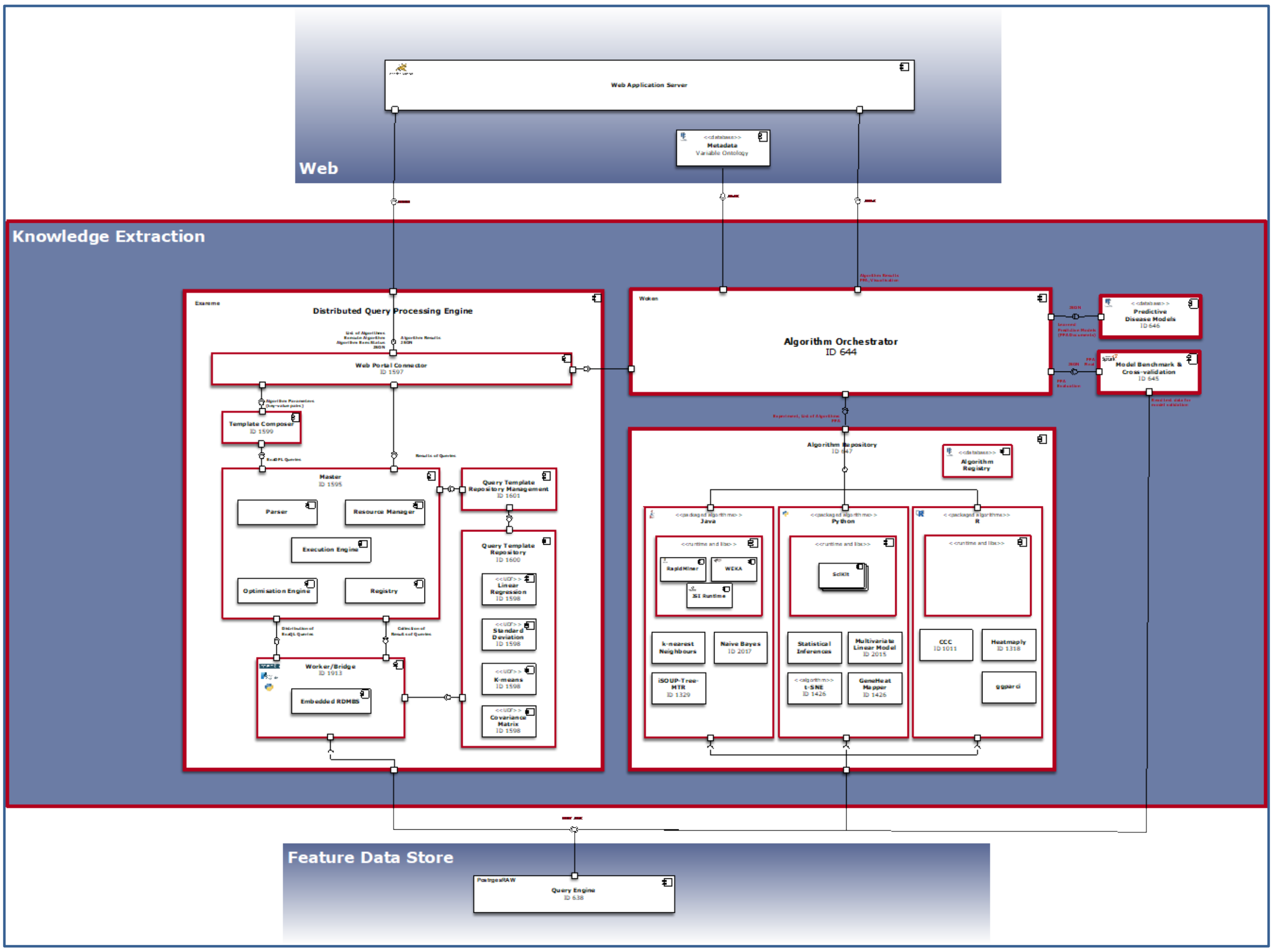


Figure 32: Knowledge Extraction Subsystem

Figure 33 depicts the interaction between the Algorithm Factory components for a typical use case of running an experiment, ordered from the MIP Web sub-system.

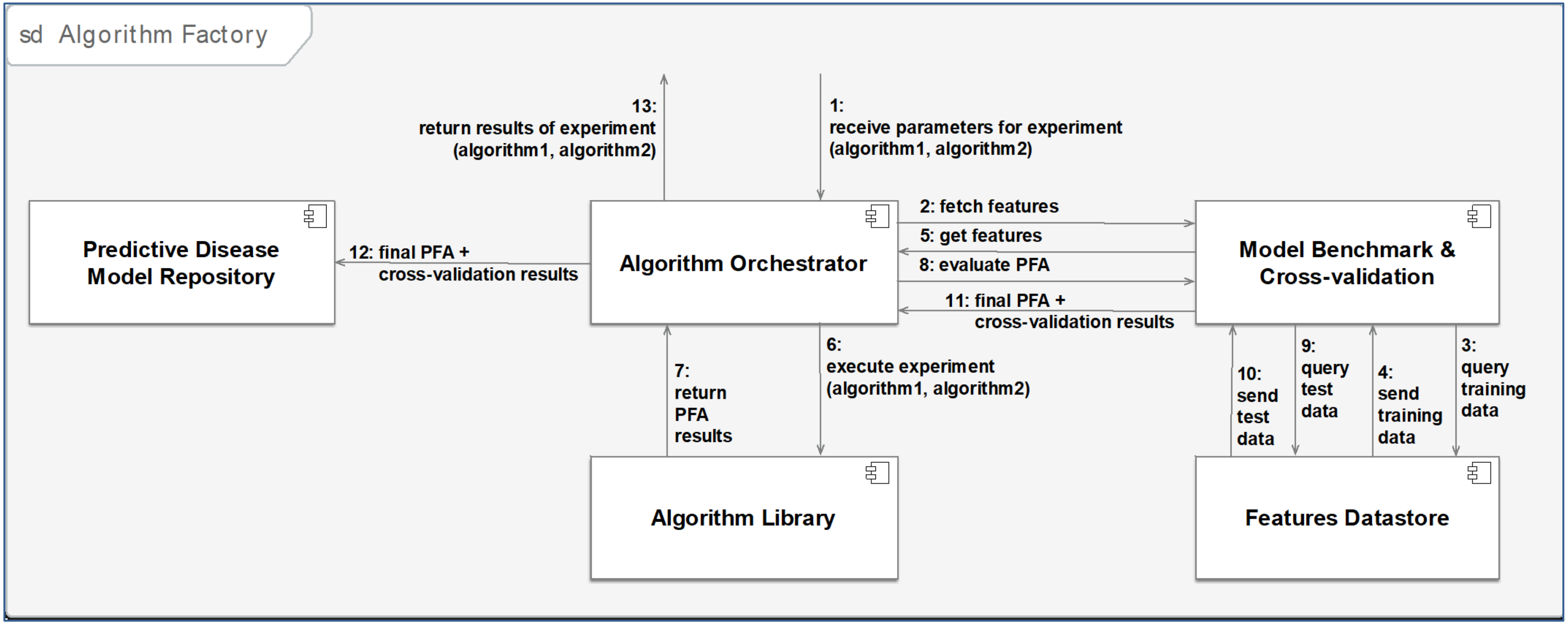

sd Algorithm Factory
13: return results of experiment (algorithm1, algorithm2)
1: receive parameters for experiment (algorithm1, algorithm2)
Predictive Disease Model Repository
12: final PFA + cross-validation results
Algorithm Orchestrator
2: fetch features
5: get features
8: evaluate PFA
11: final PFA + cross-validation results
Model Benchmark & Cross-validation
7: return PFA results
6: execute experiment (algorithm1, algorithm2)
10: send test data
9: query test data
4: send training data
3: query training data
Algorithm Library
Features Datastore

Figure 33: Algorithm Factory Communication Diagram

Predictive Disease Model Repository

This component serves as a permanent storage and search service for trained PFA models and their predictive error estimates.

3.1.4.2 Distributed Query Engine - Exareme

The Distributed Query Processing Engine plays a role in the Knowledge Extraction sub-system of the MIP platform. Master components deployed in the central federation node communicate with workers deployed in each of the hospitals, on one side, and with the Web sub-system components, on the other side. The Distributed Query Processing Engine does not allow direct communication between workers in different hospitals. Worker components, deployed in the hospitals, fetch the data from the local Feature Tables in the Features Data Store sub-system using the REST API and transfer the data to the master component for aggregation.

Systems Overview

The Distributed Query Engine, based on the open source project Exareme, is used as a traditional database system for: (1) data definition (creating, modifying, and removing database objects such as tables, indexes, and users), (2) data manipulation (data querying), and (3) external data import (from files or other databases). It is a distributed relational database management system extended with the support for complex field types – JSON, CSV and TSV.

The Distributed Query Engine uses a proprietary data manipulation language ExaDFL for specifying and orchestration of data processing. The Distributed Query Engine organises data processing in workflows designed as direct acyclic graphs (DAGs) – relational query operators are graph vertices, and the data flows between the operators are graph edges. ExaDFL is based on SQL extended with user-defined functions (UDFs) and data parallelism primitives. User-defined functions are used for specifying local data processing workflows and performing complex calculations on distributed data set partitions. ExaDFL primitives that support parallelism are declarative statements supporting parallel execution of partial queries on partitioned data sets.

The Distributed Query Engine translates ExaDFL queries to its internal declarative data manipulation language ExaQL, based on SQL-92 with extensions, for execution of query operators and user-defined functions on the distributed data set partitions.

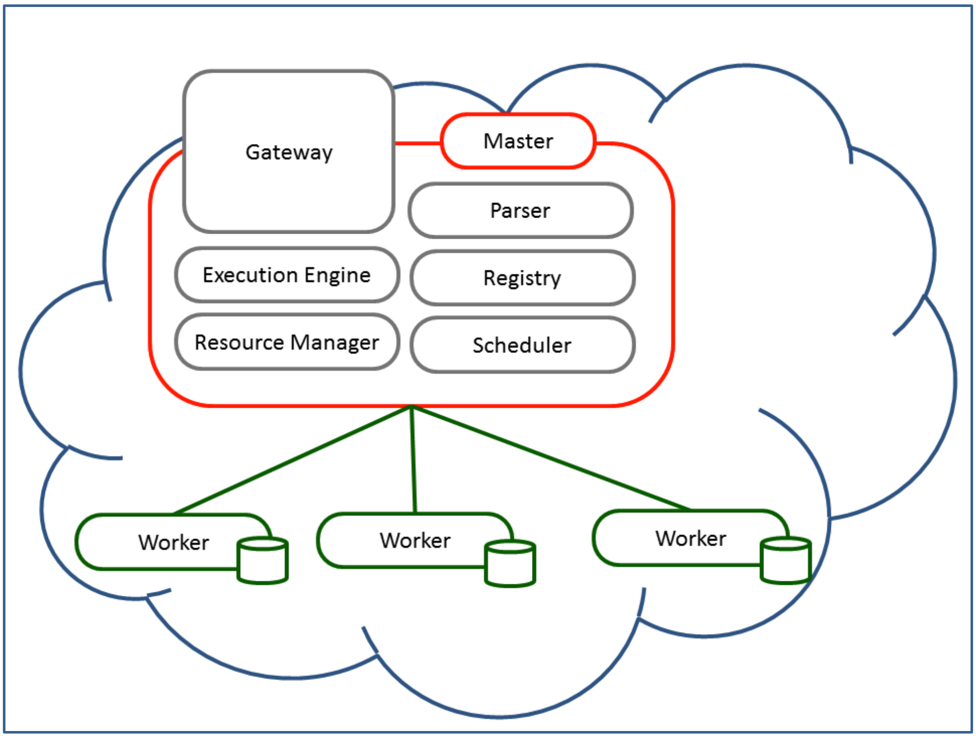


Figure 34: Distributed Query Engine Architecture Overview

The three main components of the Distributed Query Engine are:

1) **Worker** – an embedded SQLite relational database management system with Another Python SQLite Wrapper (APSW) – a Python API for SQLite running on the local hospital execution environments. It fetches local Feature Data Set sub-system's Features Table data set partitions needed for the execution of query operators and user defined functions and cashes those data set partitions to its local data storage for subsequent querying using ExaQL primitives. Worker is also a data processing system with functions for file import/export, keyword analysis, data mining tasks, fast indexing, pivoting, statistics and data processing workflow execution

2) **Master** – main entry point to the distributed query engine, running on the central federation execution environment, responsible for the coordination of the execution of other components. It aggregates query results transmitted by the Worker components distributed throughout the local hospital execution environments. Master consists of the following components:

- **Registry** – stores all information about the data and allocated resources, i.e. allocated data set partitions and their execution environments

- **Resource Manager** – allocates and de-allocates data processing resources on demand of the Execution Engine

- **Execution Engine** – requests allocation of resources from the Resource Manager, resolves the dependencies between the query operators to create a schedule of their execution in direct acyclic graph-oriented workflows, monitors the execution of the workflows and handles failures

- **Optimizer/Scheduler** – transforms ExaDFL queries into the distributed ExaQL statements and creates query execution plan by assigning operators to their respective workers

- **Gateway (Web Portal Connector)** – provides web for the communication between the Master component and the Web sub-system components

3) **Query Template Repository** – version-controlled source code store for the query templates in the form of User Defined Functions (UDFs). It is used both by the Worker and the Master components

Supported Data Processing Workflow Types

The source code of each algorithm is split into a set of local queries executed in parallel by the Worker components on the local data sets and one or more global processing executed by the Master component on the central federation node. The source code of each local and global data processing is written in a form of a workflow of SQL queries extended with user-defined functions. The source code is stored in .sql files in the Query Template Repository component. Supported data processing workflow types are:

1) **Local-global workflow** – local data processing executed in local execution environments, the aggregated results merged on the master node followed by additional data processing steps, if needed

2) **Multistep local-global workflows** – data processing workflow of predefined number of local-global data processing

3) **Iterative local-global workflows** – execution of the local-global data processing until a convergence criterion is reached (under development)

Algorithm Execution Steps

1) The Gateway component receives a user request for running an algorithm with submitted parameter values

2) The Template Composer fetches the stored local and global query templates needed for executing the selected algorithm from the Query Template Repository and creates an Algorithm template using ExaDFL primitives. Each algorithm template has an associated JSON

properties file that contains meta-information such as the algorithm's name, description, type, and parameters. Based on the type of the algorithm the type of the data processing workflow is determined

3) The Algorithm template that describes parameterized distributed workflows are forwarded to the Optimizer for generating the execution plan
4) The execution plan is forwarded to the Scheduler for determining partial algorithm execution plans, which are dispatched to Worker components running in local hospital execution environments
5) Each of the Workers executes the local data processing, then sends a confirmation of the successful execution to the Execution Engine in the central federation execution environment
6) Upon receiving success confirmations from all the Workers, the Scheduler determines global data processing plan and sends it to the Execution Engine
7) The Execution Engine then merges the aggregated results of all the workers, executes the global data processing plan and confirms its successful execution back to the Scheduler
8) The Scheduler checks if the complete local-global data processing plan has been completed
9) In case of the successful completion of the plan, it forwards the aggregated results to the user. In case the plan has not been completed, the Scheduler determines the next set of local data processing plans and the whole process of local-global plan execution is repeated until the successful completion of the algorithm

Overview of The Supported Features

The Distributed Query Processing Engine provides the following features:

1) List of the available algorithms
2) Requesting the execution of any of the available algorithms, and submission of relevant parameters
3) Execution status of the executing algorithms
4) Execution results of completed algorithms

The Distributed Query Processing Engine does not support automatic machine learning model validation. It does not provide out-of-the box predictive error estimation nor is there a component for recording the estimated accuracy of the trained machine learning models. The Algorithm Factory component can be used alongside the Distributed Query Processing Engine for trained model benchmarking and validation.

The MIP Distributed Query Processing Engine supports the following algorithms implemented as UDFs:

- **K-Means**
- **Linear Regression**

3.1.5 Web Subsystem

This section provides a brief overview of the functionality of the MIP Web sub-system. A detailed description of the front end functionality is provided in the MIP Web UI – User Guidelines, V2.0 Public Release:

(https://hbpmedical.github.io/documentation/HBP_SP8_UserGuide_latest.pdf).

The Web Sub-system provides a web portal and web applications for the end-users of the Platform. Users can explore only aggregated statistical data and perform data analysis using machine-learning methods provided by the Knowledge Extraction sub-system components. Web sub-system components have no direct access to the Feature Data Store sub-system where the individual patient de-identified health-related feature data are stored. For privacy reasons, the MIP allows exploration only of statistical data.

The Web Sub-system provides the following applications:

- **Collaboration Space** – the landing page of the Medical Informatics Platform, displaying a summary of statistics (users, available variables, written articles), and the latest three shared models and articles. It also provides a link to the Article Builder web application
- **Data Exploration –** a statistical exploration of patient feature data (i.e. variables). It is possible to explore only statistically aggregated data, not information from an individual patient. This web application provides on-the-fly generation of the descriptive statistics and contains a caching mechanism to handle any future data import in an automated way. It uses information stored in a Metadata database to display additional information about the displayed statistical data, such as data acquisition methodology, units, variable type (nominal or continuous), etc. This web application provides the functionality to search, select and classify data elements as variables, co-variables and filters for configuration of the statistical or machine learning models.
- **Model Builder –** configuration/design of statistical or predictive machine learning models. It also provides visualisation for searching data element types, select and classify data elements as variables, co-variables (nominal and continuous) and filters. Once the model is designed, a design matrix is populated with the selected data. The Model Builder provides a visual representation of the design matrix and the selected data for inspection before running a statistical, feature extraction or machine learning algorithms. It also provides an option to save the designed models
- **Experiment Builder & Disease Models** – a selection of a statistical, feature extraction or machine learning method, the configuration of the method parameters and the parameters for the trained model validation for supervised machine learning, as well as launching of the machine learning experiment. This application displays experiment validation results as bar charts and confusion matrices
- **Article Builder** – writing articles using the results of the executed experiments
- **Third-party Applications and Viewers** – a portal for accessing third-party web applications for data exploration and visualisation

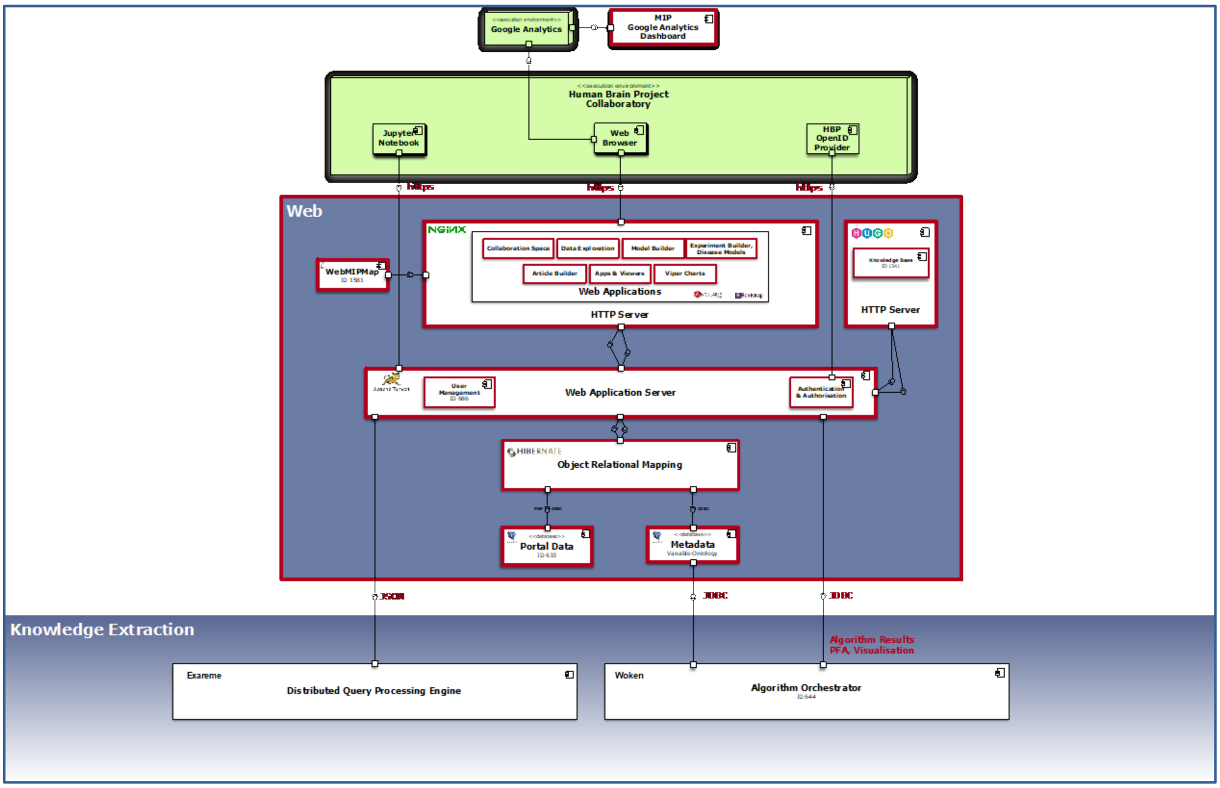


Figure 35: Web Subsystem

The Web-Sub-system allows access to its back end services and the Knowledge Extraction sub-system's Algorithm Factory through Jupyter notebooks running in the Human Brain Project's Collaboratory environment.

The Web Sub-system's Authentication and Authorisation component is integrated with the HBP Collaboratory's OpenID authentication service. The User Management component maintains an access control list and logging of user activities on the Data Exploration, Model Builder and Experiment Builder web applications.

Google Analytics Dashboard is set up for monitoring the usage of the Platform web services: tracking users and their behaviour and keeping an audit log with all user activities to detect potential Platform abuse and take preventive measures.

3.2 Deployment Architecture Overview

This section contains a brief overview of the key MIP deployment architecture concepts, relevant to understand the context of the MIP Software Installation use case specification.

A detailed description of the deployment architecture and its components is out of the scope of this document. It will be provided in the Deployment Specification document, including the following:

- Deployment model (execution environments, deployment artefacts and runtime components)
- Use case specifications (Software Installation, Data Preparation and Data Harmonisation)
- Deployment project configuration guide

3.2.1 Microservice Architecture

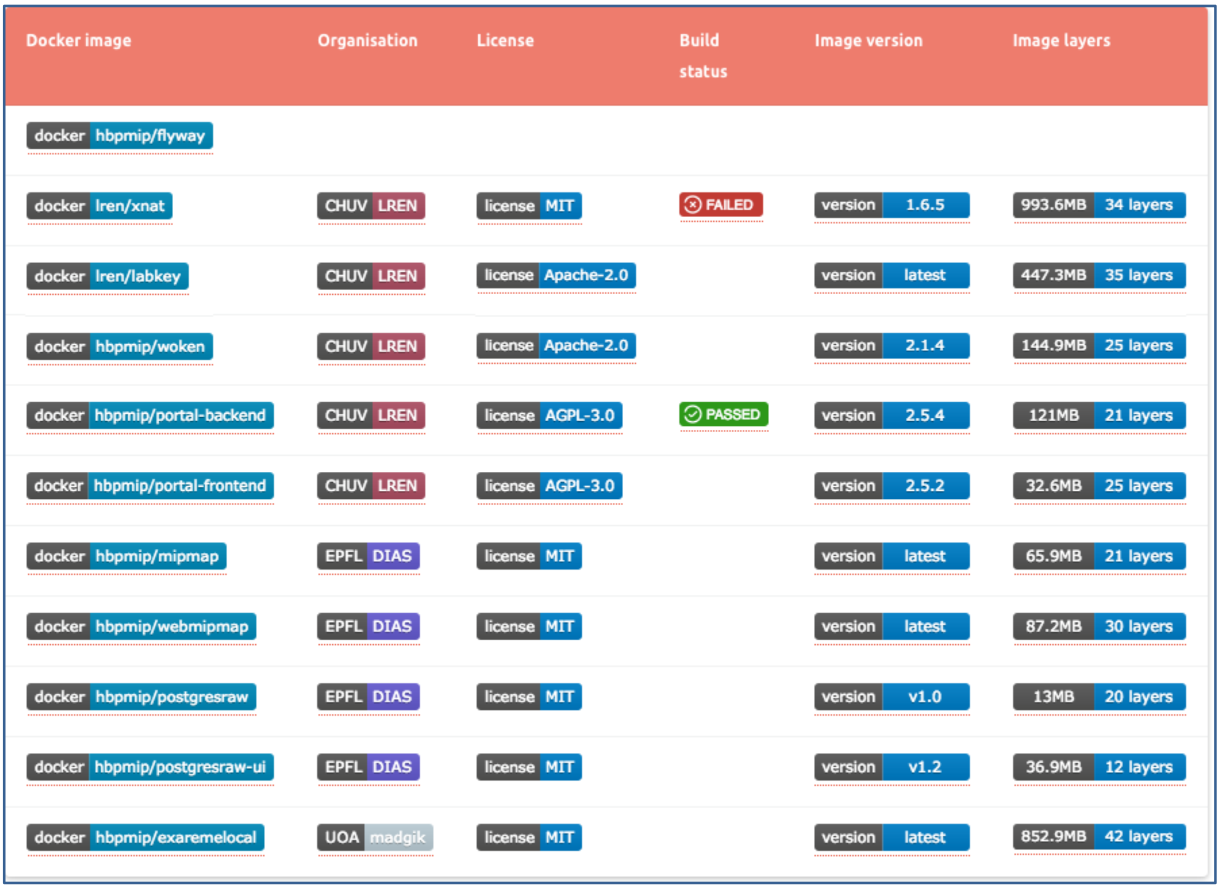
Each of the SP8 teams was focusing on delivering software components in their specific area of expertise using different technology stacks – Java, Python, R, MATLAB, Scala. As opposed to a monolithic application architecture, microservice architecture allowed the teams to work independently in their specific functional areas: Web, distributed query processing, algorithm orchestration, data-mining, statistics and machine learning algorithms, integration and verification, local data store mirror, brain scan processing and ETL, data transformation and data harmonisation.

Another significant advantage of the microservice architecture is a possibility to adopt new technology and add new features incrementally. For example, encapsulated and loosely coupled permanent data storage can be replaced with a distributed big data-ready technology packaged and deployed in Docker containers, having no impact on the surrounding data processing, ETL and analytic software components.

Operating-system level virtualisation using Docker containers on top of Linux operating system has been chosen to build and deploy microservices and run corresponding processes. Software modules are packaged as Docker images and then integrated into a production version of the distributed MIP application using continuous integration software.

3.2.2 Docker Images as Microservices

MIP software developed by the HBP partners and 3rd party software components is packaged as microservices implemented as Docker images: independently deployable, small, loosely coupled services, each one running a unique process and communicating through a well-defined lightweight mechanism. Updating a component does not require redeployments of the entire application. MIP microservice deployment architecture supports a continuous integration and continuous deployment approach.



Docker image	Organisation	License	Build status	Image version	Image layers
docker hbpmip/flyway					
docker lren/xnat	CHUV LREN	license MIT	⊗ FAILED	version 1.6.5	993.6MB 34 layers
docker lren/labkey	CHUV LREN	license Apache-2.0		version latest	447.3MB 35 layers
docker hbpmip/woken	CHUV LREN	license Apache-2.0		version 2.1.4	144.9MB 25 layers
docker hbpmip/portal-backend	CHUV LREN	license AGPL-3.0	✓ PASSED	version 2.5.4	121MB 21 layers
docker hbpmip/portal-frontend	CHUV LREN	license AGPL-3.0		version 2.5.2	32.6MB 25 layers
docker hbpmip/mipmap	EPFL DIAS	license MIT		version latest	65.9MB 21 layers
docker hbpmip/webmipmap	EPFL DIAS	license MIT		version latest	87.2MB 30 layers
docker hbpmip/postgresraw	EPFL DIAS	license MIT		version v1.0	13MB 20 layers
docker hbpmip/postgresraw-ui	EPFL DIAS	license MIT		version v1.2	36.9MB 12 layers
docker hbpmip/exaremelocal	UOA madgik	license MIT		version latest	852.9MB 42 layers

Figure 36: List of MIP Docker Images

3.2.3 Automated Installation and Configuration of MIP Software

Platform for fast deployment of services on bare metal or preconfigured virtual machines supporting clustering, security and monitoring is based on Cisco's Mantl rapid deployment project. The MIP is deployed on Mesos stack with added support for automated deployment/upgrade of services managed by Mesos Marathon and hardened security of the Ubuntu operating system. The services are built using Ansible scripts, unifying operation system configuration, middleware and application software deployment.

The MIP Hospital Deployment use cases planned for demonstration are specified in the next Chapter (MIP Software Installation Use Case Specification). Installation of the MIP in each new hospital is considered as a new git project, created as a clone of the generic Microservice Infrastructure project with configuration parameters updates tailored to a specific new hospital execution environment. Generic automatic MIP installation and configuration is stored and documented here:

https://github.com/HBPMedical/mip-microservices-infrastructure

3.2.4 MIP Software Installation Use Case Specification

The MIP microservice deployment architecture enables agile continuous integration and continuous deployment of components developed or modified by different European-wide teams. This architecture enables efficient future upgrades of the Platform with new technologies and new features needed to support evolved clinical needs. Automation of configuration and installation of the MIP software minimises IT efforts to keep the maximum focus on the scientific and clinical aspects of the projects.

Table 5 - Use Case Specification: Medical Informatics Platform Software Installation

Actors		HIT: Hospital IT Engineer MIT: MIP Deployment Engineer
Use Case Objective		Installation of the Medical Informatics Platform hardware and software in a hospital data centre
Pre-conditions		1) Formally approved investment in infrastructure, software, time and material 2) Signed Medical Informatics Platform Deployment and Evaluation Agreement 3) Infrastructure, software, time and material procured by hospital
Main Flow of Events		
Event ID	**Actor ID**	**Event Description**
E01	**HIT**	Prepare data centre for installing and configuring new MIP servers, storage and network

E02	**HIT**	Install MIP all-in-one server or separate servers (typical hospital configuration is provided below): a. Data capture and de-identification server CPU: 2-core x64; RAM: 2 GB; Storage: 50 GB; Security level: Highly secure clinical network b. Pre-processing server CPU: 12-core x64; RAM: 32 GB; Storage: 16 TB; Security Level: Secure research network c. Knowledge extraction and web server CPU: 8-core x64; RAM: 32 GB; Storage: 2 TB; Security Level: Secure research network or DMZ
E03	**HIT**	Install operating system on MIP servers • recommendation: Ubuntu 16.04 LTS / RHEL 7.2+ / CentOS 7.2+
E04	**HIT**	Provide sudo access rights for each MIP server to MIP deployment engineer
E05	**HIT**	Configure IPv4/IPv6 settings for each MIP server
E06	**HIT**	Configure SSH VPN tunnelling for remote connection with the MIP deployment team environment a. Install and run OpenSSH server on each MIP server b. Configure TCP port 22 for ingress SSH traffic on each MIP server c. Open port 22 for ingress traffic through firewall(s) between each MIP server and the Internet
E07	**HIT**	Configure TCP port 443 for egress HTTPS traffic on MIP servers and open port in firewall(s) for: a. Software package repositories (Ubuntu, Mesosphere, PyPI) b. Source code repositories (GitHub, Bitbucket, Launchpad, CHUV git) c. Docker registries (Docker Hub, CHUV private Docker registry)
E08	**MIT**	Install and configure MIP software automatically, using Ansible: a. Clone the generic Microservice Infrastructure project to create a git project for storing the new MIP environment configuration b. Prepare a configuration for automatic installation: • Install Python2 on the MIP servers – Ansible requires Python2 to run • Install MATLAB 2016b – required by SPM software for neuromorphometric processing • Server names and TCP/IP configuration c. Store the configuration in git, encrypt the passwords and confidential information d. Run a single Ansible script for the new MIP installation and configuration to: • Install middleware – libraries, runtimes, DBMSs and open source software • Deploy Docker images with software developed by MIP teams
E09	**MIT**	Confirm that all the processes are up and running from Marathon administrator's dashboard

E10	**MIT**	Backup the installation and configuration scripts on external server: • MIP team uses a private storage space on Bitbucket.org • Using the private repository, it is possible to safely and securely backup work, share it with other members of MIP for code review and receive upgrades of the platform
E11	**MIT**	Configure MIP backup for each MIP server in standard data centre backup environment
Special Requirements		Open relevant ports on firewalls, subject to the specific hospital IT security configuration
Post-conditions		1) MIP software is installed on all servers with all processes up and running 2) MIP platform is ready for data processing, storing and analysis
Scientific Added-value		1) The hospital data centre has a centralised platform for processing, storing and analysing de-identified and harmonised neuroimaging, neuropsychological, biological and demographic data of its patient population 2) Efficient, configurable and automated end-to-end software installation, unifying operation system configuration, middleware installation and microservice building minimises IT efforts to keep the focus on using the MIP platform for the scientific and clinical activities

4. MIP Product Structure

This section describes the version-controlled MIP product structure at the end of the SGA1 project phase and provides a detailed list of all the components. The MIP components are grouped in five groups, as illustrated in Figure 37.

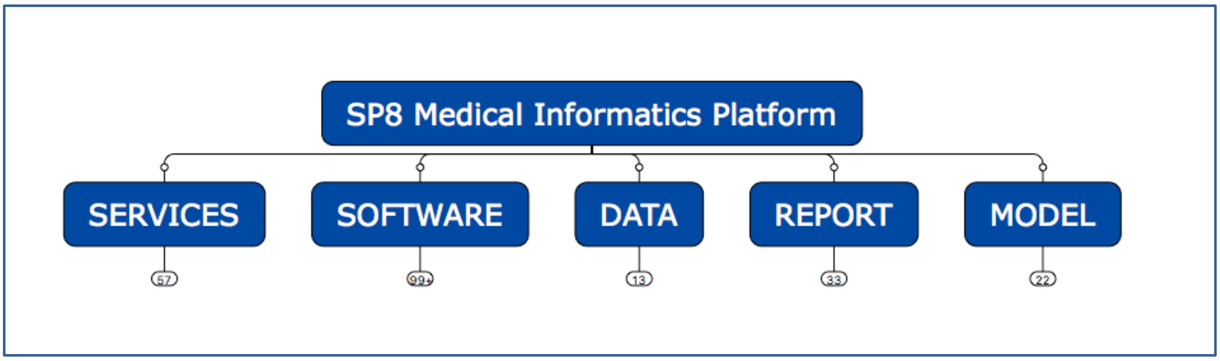


Figure 37: MIP Component Groups

The services component group contains artefacts of the research and development project activities other than software, data and models. The software group contains software components. The data group contains version controlled hospital data, metadata and test data. The report group contains communication and project management artefacts. The model group contains data models, such as disease models, reference brain models and patient de-identification profiles.

Within each of the MIP product structure groups, individual components are classified in the packages, i.e. building blocks, as illustrated in Figure 38.

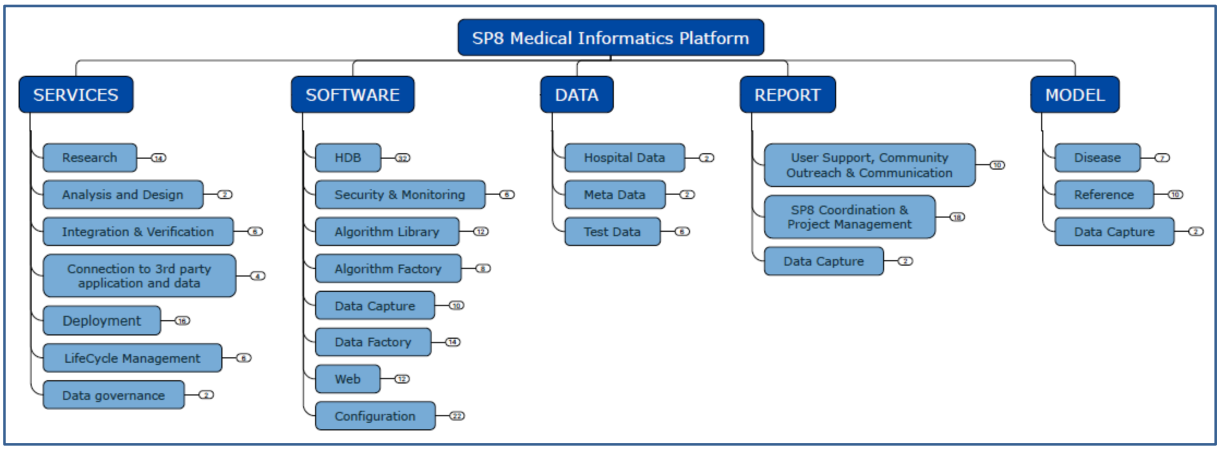


Figure 38: MIP Component Packages

The following five Figures (Figure 39, Figure 40, Figure 41, Figure 42, Figure 43) give a detailed breakdown of the component packages into individual components.

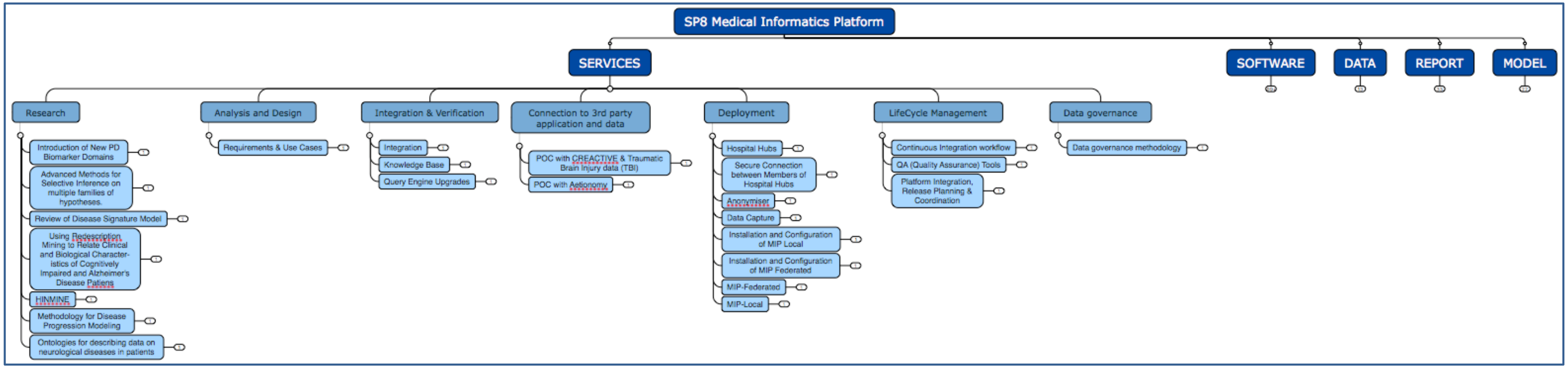

SP8 Medical Informatics Platform
SERVICES
SOFTWARE
DATA
REPORT
MODEL
Research
Introduction of New PD Biomarker Domains
Advanced Methods for Selective Inference on multiple families of hypotheses.
Review of Disease Signature Model
Using Redescription Mining to Relate Clinical and Biological Characteristics of Cognitively Impaired and Alzheimer's Disease Patiens
HINMINE
Methodology for Disease Progression Modeling
Ontologies for describing data on neurological diseases in patients
Analysis and Design
Requirements & Use Cases
Integration & Verification
Integration
Knowledge Base
Query Engine Upgrades
Connection to 3rd party application and data
POC with CREACTIVE & Traumatic Brain Injury data (TBI)
POC with Aetionomy
Deployment
Hospital Hubs
Secure Connection between Members of Hospital Hubs
Anonymiser
Data Capture
Installation and Configuration of MIP Local
Installation and Configuration of MIP Federated
MIP-Federated
MIP-Local
LifeCycle Management
Continuous Integration workflow
QA (Quality Assurance) Tools
Platform Integration, Release Planning & Coordination
Data governance
Data governance methodology

Figure 39: MIP Services Component Structure

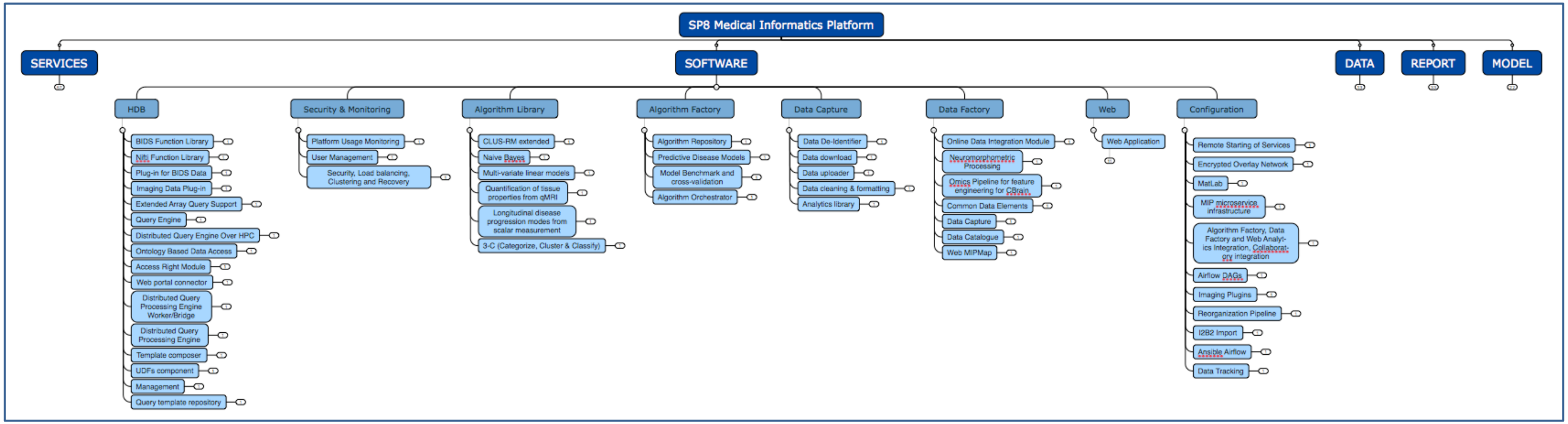

SP8 Medical Informatics Platform
SERVICES
SOFTWARE
DATA
REPORT
MODEL
HDB
BIDS Function Library
Nifti Function Library
Plug-in for BIDS Data
Imaging Data Plug-in
Extended Array Query Support
Query Engine
Distributed Query Engine Over HPC
Ontology Based Data Access
Access Right Module
Web portal connector
Distributed Query Processing Engine Worker/Bridge
Distributed Query Processing Engine
Template composer
UDFs component
Management
Query template repository
Security & Monitoring
Platform Usage Monitoring
User Management
Security, Load balancing, Clustering and Recovery
Algorithm Library
CLUS-RM extended
Naive Bayes
Multi-variate linear models
Quantification of tissue properties from qMRI
Longitudinal disease progression modes from scalar measurement
3-C (Categorize, Cluster & Classify)
Algorithm Factory
Algorithm Repository
Predictive Disease Models
Model Benchmark and cross-validation
Algorithm Orchestrator
Data Capture
Data De-Identifier
Data download
Data uploader
Data cleaning & formatting
Analytics library
Data Factory
Online Data Integration Module
Neuromorphometric Processing
Omics Pipeline for feature engineering for CBrain
Common Data Elements
Data Capture
Data Catalogue
Web MIPMap
Web
Web Application
Configuration
Remote Starting of Services
Encrypted Overlay Network
MatLab
MIP microservice infrastructure
Algorithm Factory, Data Factory and Web Analyt-ics Integration, Collaborat-ory integration
Airflow DAGs
Imaging Plugins
Reorganization Pipeline
I2B2 Import
Ansible Airflow
Data Tracking

Figure 40: MIP Software Component Structure

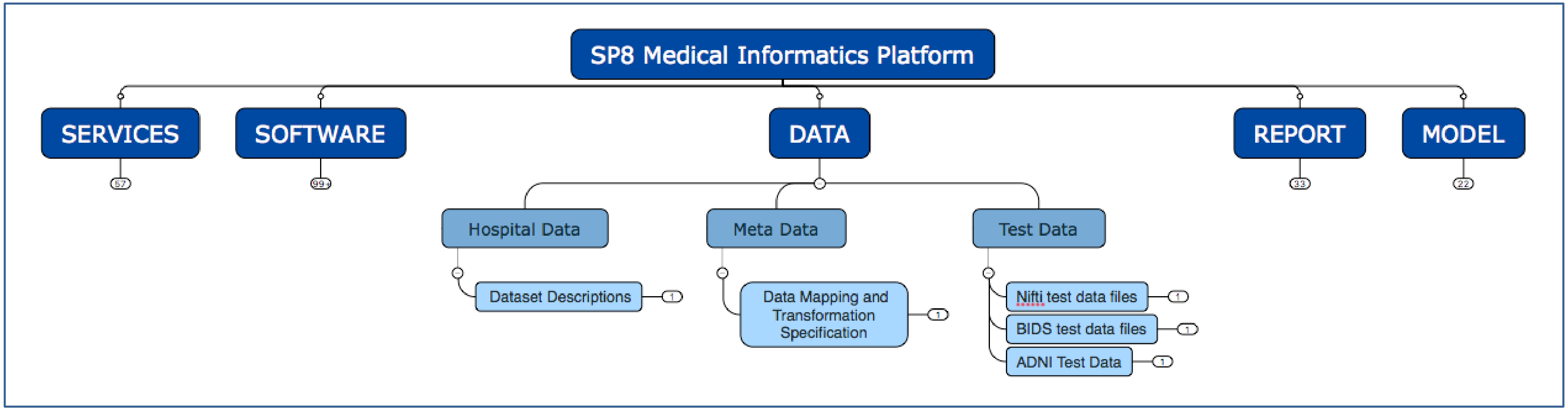


Figure 41: MIP Data Component Structure

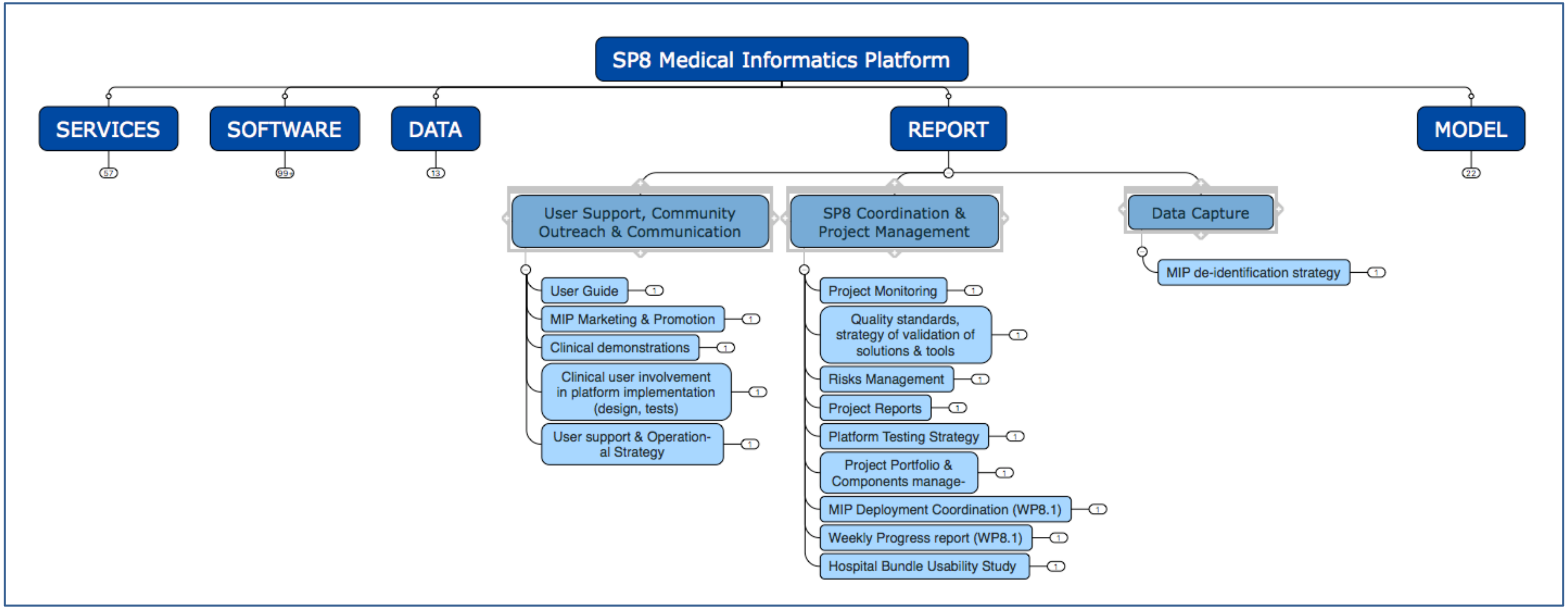

SP8 Medical Informatics Platform
SERVICES
57
SOFTWARE
99+
DATA
13
REPORT
MODEL
22
User Support, Community Outreach & Communication
User Guide
MIP Marketing & Promotion
Clinical demonstrations
Clinical user involvement in platform implementation (design, tests)
User support & Operation-al Strategy
SP8 Coordination & Project Management
Project Monitoring
Quality standards, strategy of validation of solutions & tools
Risks Management
Project Reports
Platform Testing Strategy
Project Portfolio & Components manage-
MIP Deployment Coordination (WP8.1)
Weekly Progress report (WP8.1)
Hospital Bundle Usability Study
Data Capture
MIP de-identification strategy
1

Figure 42: MIP Report Component Structure

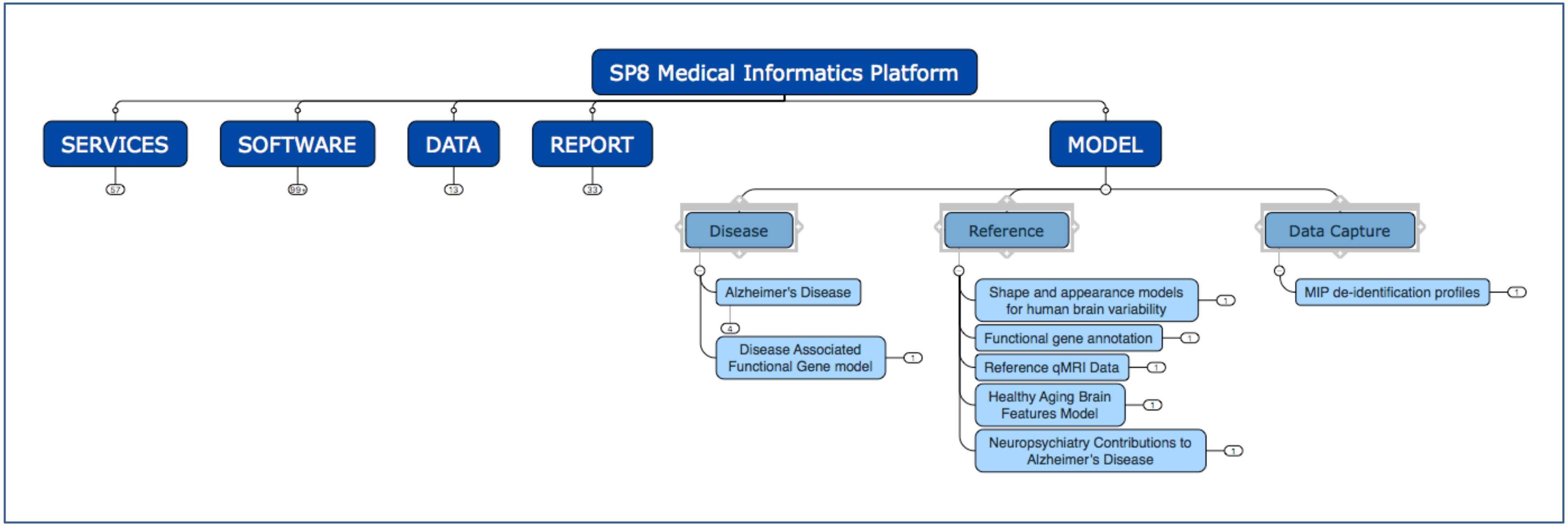


Figure 43: MIP Model Component Structure

4.1 Software Components

The software group contains software packages and corresponding components.

Table 6: MIP Software Components

building block	component	ID	task	WP
Algorithm Factory	Algorithm Repository	647	T8.5.2	WP8.5
Algorithm Factory	Predictive Disease Models	646	T8.5.2	WP8.5
Algorithm Factory	Model Benchmark and cross-validation	645	T8.5.2	WP8.5
Algorithm Factory	Algorithm Orchestrator	2938	T8.5.2	WP8.5
Algorithm Library	CLUS-RM extended	1329	T8.3.5	WP8.3
Algorithm Library	Naive Bayes	2017	T8.4.2	WP8.4
Algorithm Library	Multi-variate linear models	2015	T8.4.2	WP8.4
Algorithm Library	Quantification of tissue properties from qMRI	1287	T8.4.3	WP8.4
Algorithm Library	Longitudinal disease progression modes from scalar measurement	2416	T8.3.12	WP8.3
Algorithm Library	3-C (Categorize, Cluster & Classify)	1011	T8.3.1	WP8.3
Algorithm Library	Web Application > Heatmaply	1318	T8.3.2	WP8.3
Configuration	Remote Starting of Services	1759	T8.1.3	WP8.1
Configuration	Encrypted Overlay Network	1760	T8.1.3	WP8.1
Configuration	MatLab	665	T8.5.2	WP8.5
Configuration	MIP microservice infrastructure	102	T8.5.2	WP8.5
Configuration	Algorithm Factory, Data Factory and Web Analytics Integration, Collaboratory integration	2939	T8.5.2	WP8.5
Configuration	Airflow DAGs	664	T8.5.2	WP8.5
Configuration	Imaging Plugins	2929	T8.5.2	WP8.5
Configuration	Reorganization Pipeline	2930	T8.5.2	WP8.5
Configuration	I2B2 Import	2931	T8.5.2	WP8.5
Configuration	Ansible Airflow	2932	T8.5.2	WP8.5
Configuration	Data Tracking	2928	T8.5.2	WP8.5
Data Capture	Data De-Identifier	2940	T8.1.1	WP8.1

building block	component	ID	task	WP
Data Capture	Data download	2865	T8.4.5	WP8.4
Data Capture	Data uploader	2862	T8.4.5	WP8.4
Data Capture	Data cleaning & formatting	2863	T8.4.5	WP8.4
Data Capture	Analytics library	2864	T8.4.5	WP8.4
Data Factory	Omics Pipeline for feature engineering for Cbrain	670	T8.5.2	WP8.5
Data Factory	Online Data Integration Module	1580	T8.1.4	WP8.1
Data Factory	Neuromorphometric Processing	671	T8.5.2	WP8.5
Data Factory	Common Data Elements	669	T8.5.2	WP8.5
Data Factory	Data Capture	2926	T8.5.2	WP8.5
Data Factory	Data Catalogue	2927	T8.5.2	WP8.5
Data Factory	WebMIPMap	1581	T8.1.4	WP8.1
HDB	BIDS Function Library	1754	T8.1.1	WP8.1
HDB	Nifti Function Library	1753	T8.1.1	WP8.1
HDB	Plug-in for BIDS Data	1752	T8.1.1	WP8.1
HDB	Imaging Data Plug-in	1751	T8.1.1	WP8.1
HDB	Extended Array Query Support	1750	T8.1.1	WP8.1
HDB	Query Engine	638	T8.1.1	WP8.1
HDB	Distributed Query Engine Over HPC	1755	T8.1.2	WP8.1
HDB	Ontology Based Data Access	1579	T8.1.4	WP8.1
HDB	Access Right Module	1578	T8.1.4	WP8.1
HDB	Web portal connector	1597	T8.1.5	WP8.1
HDB	Distributed Query Processing Engine Worker/Bridge	1596	T8.1.5	WP8.1
HDB	Distributed Query Processing Engine Master	1595	T8.1.5	WP8.1
HDB	Template composer	1599	T8.1.6	WP8.1
HDB	UDFs component	1598	T8.1.6	WP8.1
HDB	Management	1601	T8.1.7	WP8.1

building block	component	ID	task	WP
HDB	Query template repository	1600	T8.1.7	WP8.1
Security & Monitoring	Platform Usage Monitoring	685	T8.5.1	WP8.5
Security & Monitoring	User Management	686	T8.5.1	WP8.5
Security & Monitoring	Security, Load balancing, Clustering and Recovery Services	684	T8.5.2	WP8.5
Web	Web Application > Knowledge Base > Research Dataset List	2286	T8.2.2	WP8.2
Web	Web Application > Brain insight > GeneHeatMapper	1426	T8.3.10	WP8.3
Web	Web Application > Portal DB (articles, experiments, models)	633	T8.2.3	WP8.2
Web	Web Application > Knowledge Base	1541	T8.2.3	WP8.2

4.2 Service Components

The services component group contains artefacts of research and development project activities other than software, data and models.

Table 7: MIP Service Components

building block	component	ID	task	WP
Analysis and Design	Requirements & Use Cases	690	T8.6.1	WP8.6
Connection to 3rd party application and data	POC with CREACTIVE & Traumatic Brain Injury data (TBI)	1560	T8.5.2	WP8.5
Connection to 3rd party application and data	POC with Aetionomy	1562	T8.5.2	WP8.5
Data governance	Data governance methodology	687	T8.6.2	WP8.6
Deployment	Hospital Hubs	1757	T8.1.2	WP8.1
Deployment	Secure Connection between Members of Hospital Hubs	1756	T8.1.2	WP8.1
Deployment	Anonymiser	1816	T8.1.3	WP8.1
Deployment	Data Capture	1815	T8.1.3	WP8.1
Deployment	Installation and Configuration of MIP Local	1817	T8.1.3	WP8.1

building block	component	ID	task	WP
Deployment	Installation and Configuration of MIP Federated	1818	T8.1.3	WP8.1
Deployment	MIP-Federated	1557	T8.6.2	WP8.6
Deployment	MIP-Local	1556	T8.6.2	WP8.6
Integration & Verification	Knowledge Base	617	T8.5.1	WP8.5
Integration & Verification	Integration	1819	T8.1.3	WP8.1
Integration & Verification	Query Engine Upgrades	1820	T8.1.3	WP8.1
LifeCycle Management	Continuous Integration workflow	1551	T8.5.1	WP8.5
LifeCycle Management	QA (Quality Assurance) Tools	1552	T8.5.2	WP8.5
LifeCycle Management	Platform Integration, Release Planning & Coordination	1555	T8.5.3	WP8.5
Research	Introduction of New PD Biomarker Domains	1021	T8.3.1	WP8.3
Research	Advanced Methods for Selective Inference on multiple families of hypotheses.	1319	T8.3.3	WP8.3
Research	Review of Disease Signature Model	1321	T8.3.4	WP8.3
Research	Using Redescription Mining to Relate Clinical and Biological Characteristics of Cognitively Impaired and Alzheimer's Disease Patients	1323	T8.3.6	WP8.3
Research	HINMINE	1325	T8.3.7	WP8.3
Research	Methodology for Disease Progression Modelling	1424	T8.3.8	WP8.3
Research	Ontologies to describe neurological disease patient data	1331	T8.3.9	WP8.3

4.3 Data Components

The data group contains version controlled hospital data, metadata and test data.

Table 8: MIP Data Components

building block	component	ID	task	WP
Hospital Data	Dataset Descriptions	455	T8.2.1	WP8.2

building block	component	ID	task	WP
Meta Data	Data Mapping and Transformation Specification	587	T8.2.1	WP8.2
Test Data	Nifti test data files	1734	T8.1.1	WP8.1
Test Data	BIDS test data files	1733	T8.1.1	WP8.1
Test Data	ADNI Test Data	2716	T8.5.2	WP8.5

4.4 Model Components

The model group contains data models, such as disease models, reference brain models and patient de-identification profiles.

Table 9: MIP Model Components

building block	component	ID	task	WP
Data Capture	MIP de-identification profiles	2936	T8.1.3	WP8.1
Disease	Alzheimer's Disease > Longitudinal ADNI Dataset	1013	T8.3.1	WP8.3
Disease	Alzheimer's Disease > Brain Features Model	608	T8.4.1	WP8.4
Disease	Disease Associated Functional Gene model	1605	T8.4.1	WP8.4
Reference	Shape and appearance models for human brain variability	2171	T8.3.11	WP8.3
Reference	Functional gene annotation	1604	T8.4.1	WP8.4
Reference	Reference qMRI Data	1305	T8.4.3	WP8.4
Reference	Healthy Aging Brain Features Model	611	T8.4.1	WP8.4
Reference	Neuropsychiatry Contributions to Alzheimer’s Disease	2018	T8.4.2	WP8.4

5. MIP Data Management

Software architecture-related prerequisites for cross-centre data analytics are:

- a hybrid community and private execution environment deployment model
- a microservice architecture coupled with continuous integration and continuous deployment technology
- a distributed patient data storage and federated algorithm execution

This distributed, patient privacy preserving software architecture is a necessary but not sufficient condition for multi-dataset clinical studies comprising patient data from hospitals and open research cohort datasets. The datasets have overlapping data types but different ontological representations. Data is described, stored and formatted in different data structures. For execution of multi-dataset analytics, data models need to be harmonised in a common MIP data model, which is, together with dataset-specific data models, shared and synchronised between the distributed private hospital instances and community execution environment (Figure 1).

Data model harmonisation is, therefore, a key technology enabler for cross-centre multiple dataset clinical studies. It is a well-defined process supported by the application ontology software architecture, and the organisation, which establishes and maintains the rules and controls quality and integrity of the data harmonisation process.

The Data Governance and Data Selection (DGDS) committee is a centrally coordinated MIP organisational entity responsible for establishing and maintaining data governance methodology and data harmonisation rules. The members of the DGDS committee are MIP software architects, members of the expert medical committee consisting of medical doctors and clinical researchers of participating hospitals and institutes, and data managers, from the participating hospitals and the MIP R&D team.

Data harmonisation and re-harmonisation is an on-going process. With the introduction of a new dataset the whole process has to be repeated, starting with the analysis of the incoming dataset and ending with the synchronisation of (re-)harmonised data models across the distributed MIP ecosystem.

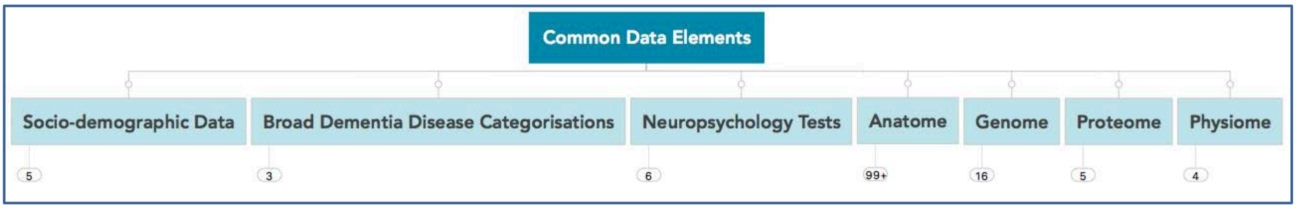


Figure 44: MIP SGA1 Common Data Elements

A diagnostic framework, i.e. the type and categorisation of diagnoses is typically hospital specific. Each hospital has its own naming classification of diseases. It is usually based on a standard classification, like ICD-10, but often a more detailed classification is needed for some disease domains. In case of dementia disorders, for example, CHRU Lille has adopted the recommendation of the Banque National Alzheimer (BNA). The CHUV in Lausanne has recently provided their adaptation of the BNA disease classification, which is planned for integration in the next release of the MIP. System validation has been based on the old ICD-10 classification. To have multi-dataset analytics involved in diagnoses, the MIP has introduced three broad disease categories – Alzheimer’s Disease, Parkinson’s Disease and Neurodegenerative Diseases. These broad disease categories are mapped to the disease definitions of each of the disease frameworks of the participating hospitals.

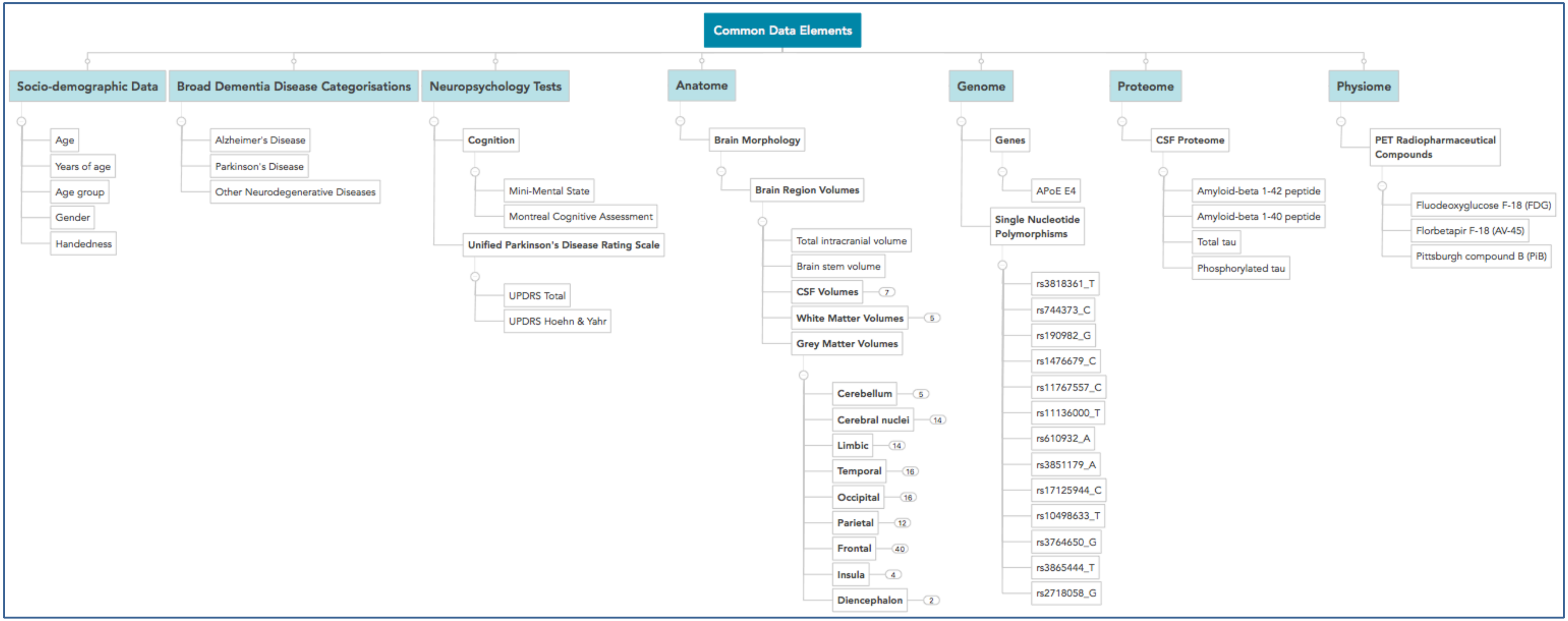


Figure 45: MIP SGA1 Common Data Element Taxonomy[1]

[1] **Note:** Brain Volume group contains 135 data elements representing brain regions classified using the standard brain anatomy classification

Table 10: High-level data model harmonisation process description

Activity Number	Activity	Description
1	Analysis of the new dataset	Initial profiling of the original de-identified patient data exported from EHR's and research datasets in CSV format, stored in clinical research data warehouses or other OLAP systems (for example I2B2). Analysis of the brain scan dataset, including the number of scan sessions, and preliminary examination of the DICOM file header information
2	Understanding the meaning of the data	Analysis of the formats and structures of received datasets. Informal description of the original data types confirmed and approved by the originating hospital/institute experts
3	Creation of data vocabularies / application ontologies	Creation of data vocabularies / MIP application ontologies for: socio-demographic data, brain regions, genome, proteome, metabolome, physiome, phenome. Creation of a hospital-specific diagnostic framework, including mapping to MIP broad disease categories. Creation of a hospital-specific neuropsychological assessment framework
4	Re-harmonisation of the common data model	Updating of the MIP common data model in coordination with expert representatives of participating hospitals and institutes
5	Update and formal approval of the Data Mapping and Transformation Specification	Formal, version controlled specification of the harmonisation and naming rules updated and formally approved by originating hospital/institute experts, MIP medical consultants and MIP software architects
6	Integration of common and dataset-specific data models	Integration and verification of common and dataset-specific data models in the MIP testing environment. Regression testing using the open research cohort datasets
7	Synchronisation of harmonised data models across the distributed MIP ecosystem	Synchronisation of (re-)harmonised data models – common hospital/insitute-specific across the private hospital/institute MIP execution environments, common and all hospital/insitute specific in the community execution environment (architecture details in chapters 1 and 3)

6. MIP Hospital Deployment Results

During the Ramp-Up Phase of the project, nineteen European university hospitals expressed interest in providing patient datasets, deploying and evaluating Medical Informatics Platform. Deployment and Evaluation Agreements were signed with seven of them (Figure 46):

- University Hospital of Lausanne, Switzerland (CHUV)
- Regional University Hospital Center of Lille, France (CHRU Lille)
- Research and Healthcare Institute in Brescia, Italy (IRCCS Fatebenefratelli Brescia)
- Metropolitan Hospital Niguarda in Milano, Italy (ASST Grande Ospedale Metropolitano Niguarda)
- Medical Center – University of Freiburg, Germany (Universitätsklinikum Freiburg)
- Medical University of Plovdiv, Bulgaria
- Tel Aviv Sourasky Medical Center in Tel Aviv, Israel

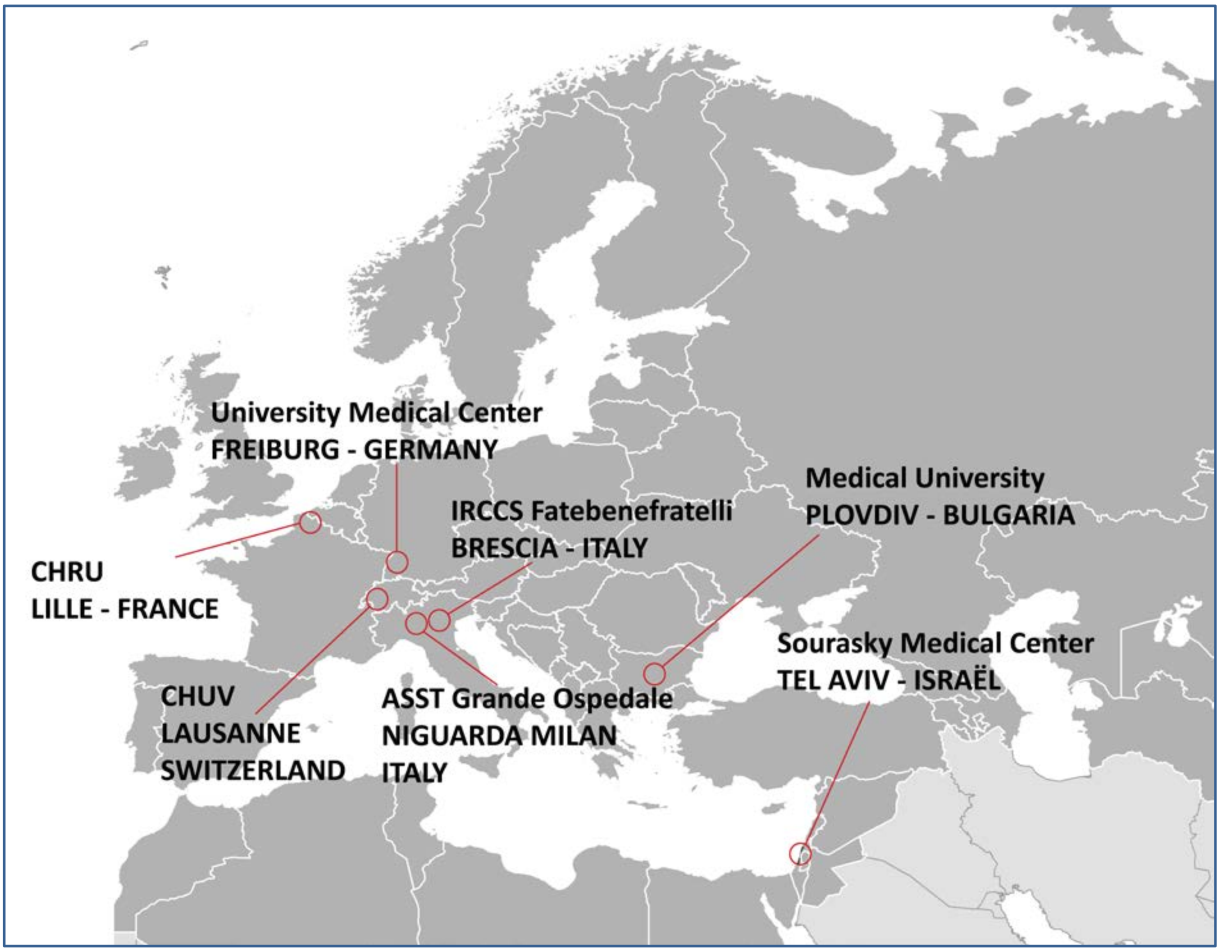


Figure 46: Deployment and Evaluation Agreements with European University Hospitals

The criteria for selecting the seven European university hospitals for providing patient datasets, deploying and evaluating the MIP are provided in Table 11:

Table 11: Criteria for selecting participating hospitals

Criteria	Description
Diversity	Hospitals in different countries. Objective: to test the MIP in different healthcare systems, using data of patients with different exposure to risk factors, disease prevalence, etc.
Size	Hospitals that have a significant number of patients and large patient datasets
Clinical Excellence	The best national hospitals with expertise in clinical neuroscience and clinical care, willingness to share data, with well-established ethics consent procedures
Available resources	Hospitals that have the personnel and IT equipment resources, and a long-term commitment to maintain the Medical Informatics Platform infrastructure
Influence	Hospitals that will promote the Medical Informatics Platform through collaboration with other hospitals in the same region or country

The Medical Informatics Platform provides support for analysing diverse biomedical and other health-relevant patient data. That includes support for multi-centre, multi-dataset studies to bridge the gap between fundamental research and clinical practice.

The scientific research significance of this project is a realistic possibility to discover hidden data patterns by combining multi-centre patient clinical datasets with available open research cohort data, such as ADNI, EDSD and PPMI, and compliance of the MIP platform concept with WHO's action areas. The Information Systems for Dementia and Dementia Research and Innovation were the reasons to select dementia and in particular Alzheimer's disease clinical study scenarios for the demonstration of MIP functionality and its scientific utility. The scientific utility of the Platform, defined as a key SP8 result, is discussed in the M24 deliverable D8.6.3.

Clinicians and clinical researchers from the three selected university hospitals have been chosen because of their expertise in the domain of dementia syndromes and of the profiles of the available patient datasets. They presented data of a significant number of patients with neurodegenerative and neurocognitive disorders, different types of dementia, high Alzheimer's disease incidence, and a variety of biological, cognitive, neuroimaging and other relevant patient data available for analytics.

Data profiles for three university hospitals from France, Italy and Switzerland, including the number of patients in each cohort dataset and the counts of patients with diagnosed Alzheimer's disease (AD), mild cognitive disorder (MCI), other neurodegenerative disorders, cognitive normal (CN) control group and not defined (N/A – not available) diagnostic are provided in Table 12.

The patient cohort dataset of the Regional University Hospital Center of Lille, France (CHRU Lille), consisted of multiple visits per patient. Data profiles in Table 12 give information about the first and the last visit that are recorded in the dataset. The patient cohort datasets of the IRCCS FBF in Brescia, Italy and the University Hospital of Lausanne (CHUV) in Switzerland consisted of a single visit per patient.

Table 12: Hospitals selected to participate in MIP system validation - data profiles[2]

Hospital	Patient Count	Recorded Visit	Diagnosis - Alzheimer's Broad Category CDE				
			AD	MCI	Other	CN	N/A
CHRU Lille France	1436	First	591	227	551	67	0
		Last	813	7	604	12	0
IRCCS FBF Brescia Italy	1960	First	151	201	192	1240	176
		Last	N/A	N/A	N/A	N/A	N/A
CHUV/CLM Lausanne Switzerland	699	First	164	78	414	41	2
		Last	N/A	N/A	N/A	N/A	N/A
ADNI	1066		222	576	0	268	0
EDSD	474		141	76	0	151	106
TOTAL	**5635**		**1269**	**1158**	**1157**	**1767**	**284**

Common data models (Figure 45) have been integrated and synchronised across the participating hospitals' private MIP execution environments. Both common and hospital-specific data models have been integrated in the central MIP community execution environment (Figure 1).

MIP SGA1 common data taxonomy is illustrated in Figure 45. Specific data taxonomies for each of the three hospitals participating in the SP8 system validation project phase are illustrated in Figure 47, Figure 48, and Figure 49.

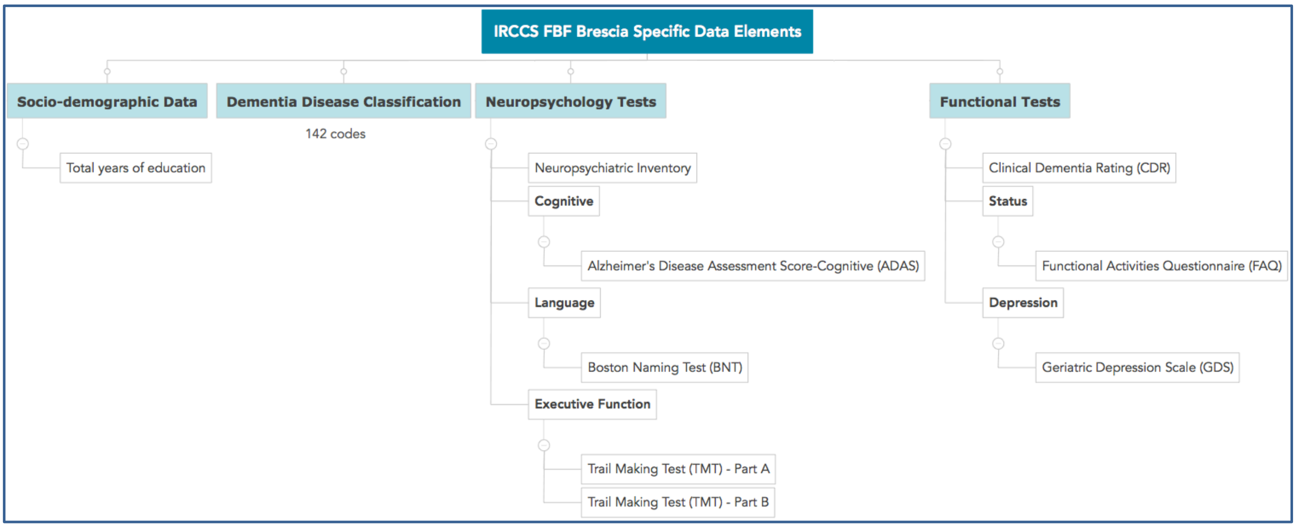

IRCCS FBF Brescia Specific Data Elements
Socio-demographic Data
Total years of education
Dementia Disease Classification
142 codes
Neuropsychology Tests
Neuropsychiatric Inventory
Cognitive
Alzheimer's Disease Assessment Score-Cognitive (ADAS)
Language
Boston Naming Test (BNT)
Executive Function
Trail Making Test (TMT) - Part A
Trail Making Test (TMT) - Part B
Functional Tests
Clinical Dementia Rating (CDR)
Status
Functional Activities Questionnaire (FAQ)
Depression
Geriatric Depression Scale (GDS)

Figure 47: IRCCS Brescia Specific Data Taxonomy

[2] **Diagnosis:** AD – Alzheimer's disease, MCI – mild cognitive impairment, CN – cognitive normal, Other – other neurodegenerative disorder, N/A – disease information not available

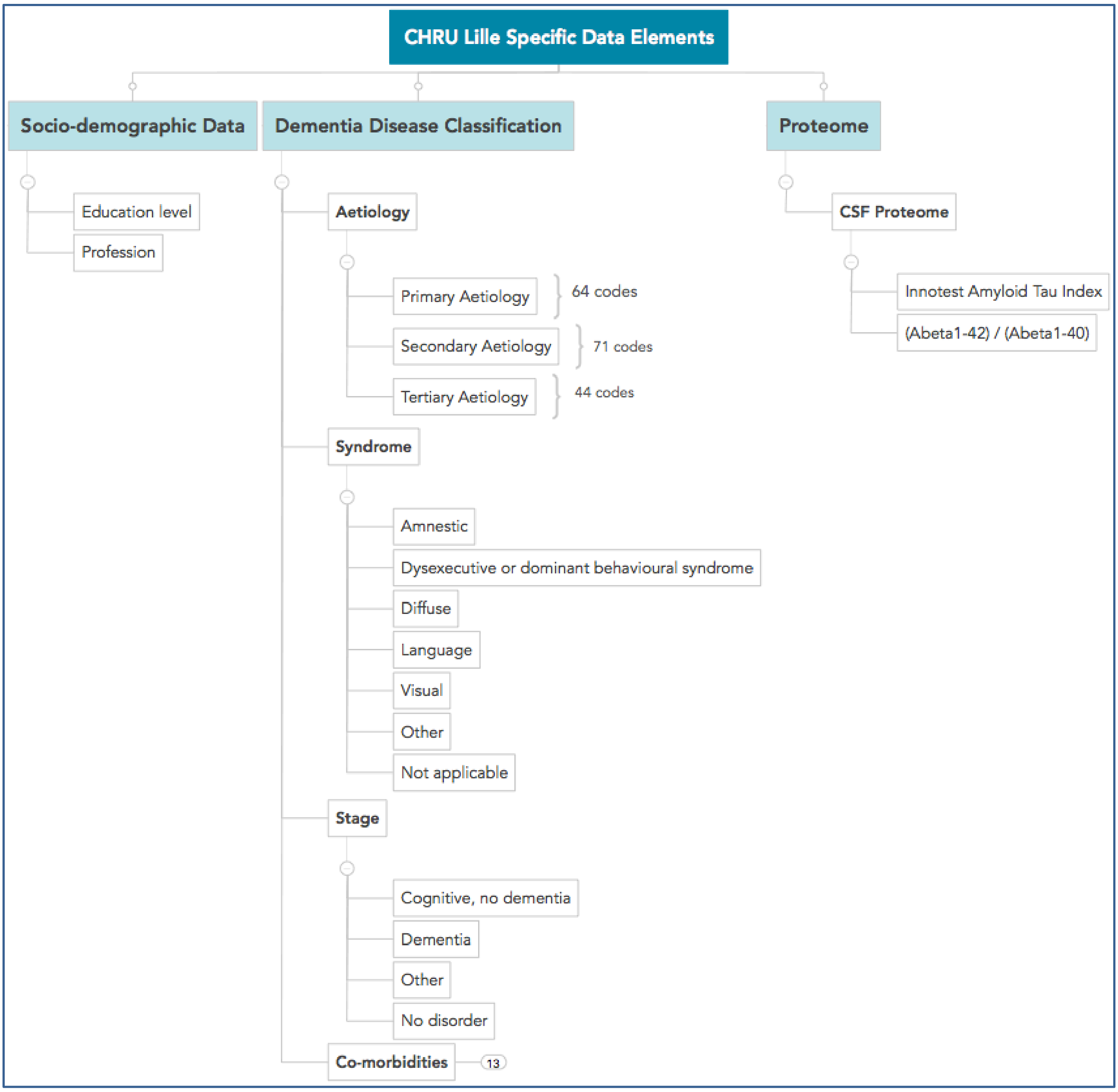


Figure 48: CHRU Lille Specific Data Taxonomy

The MIP software provides the IT prerequisites for the execution of cross-centre, multi-dataset clinical studies across the three university hospitals participating in the SP8 system validation project phase. The software harmonises and synchronises the data model. This software is privacy-preserving, it uses a hybrid community-private deployment model with centralised orchestration of statistical and machine learning algorithms.

The unsupervised machine-learning can be used, for example, to train a classifier, using CHRU-Lille data, to be able to differentiate between frontotemporal dementia and Alzheimer's disease. The learned classifier can then be applied to obtain a differential diagnosis (between frontotemporal dementia and Alzheimer's disease) in the IRCCS in Brescia and the CHUV in Lausanne. Or, we can use clinical and pathological data of deceased patients from the CHRU Lille dataset to train a machine-learning model that can be used to predict disease progression from patients in the other two hospitals. Detailed results of these studies are presented in the M24 deliverable D8.6.3.

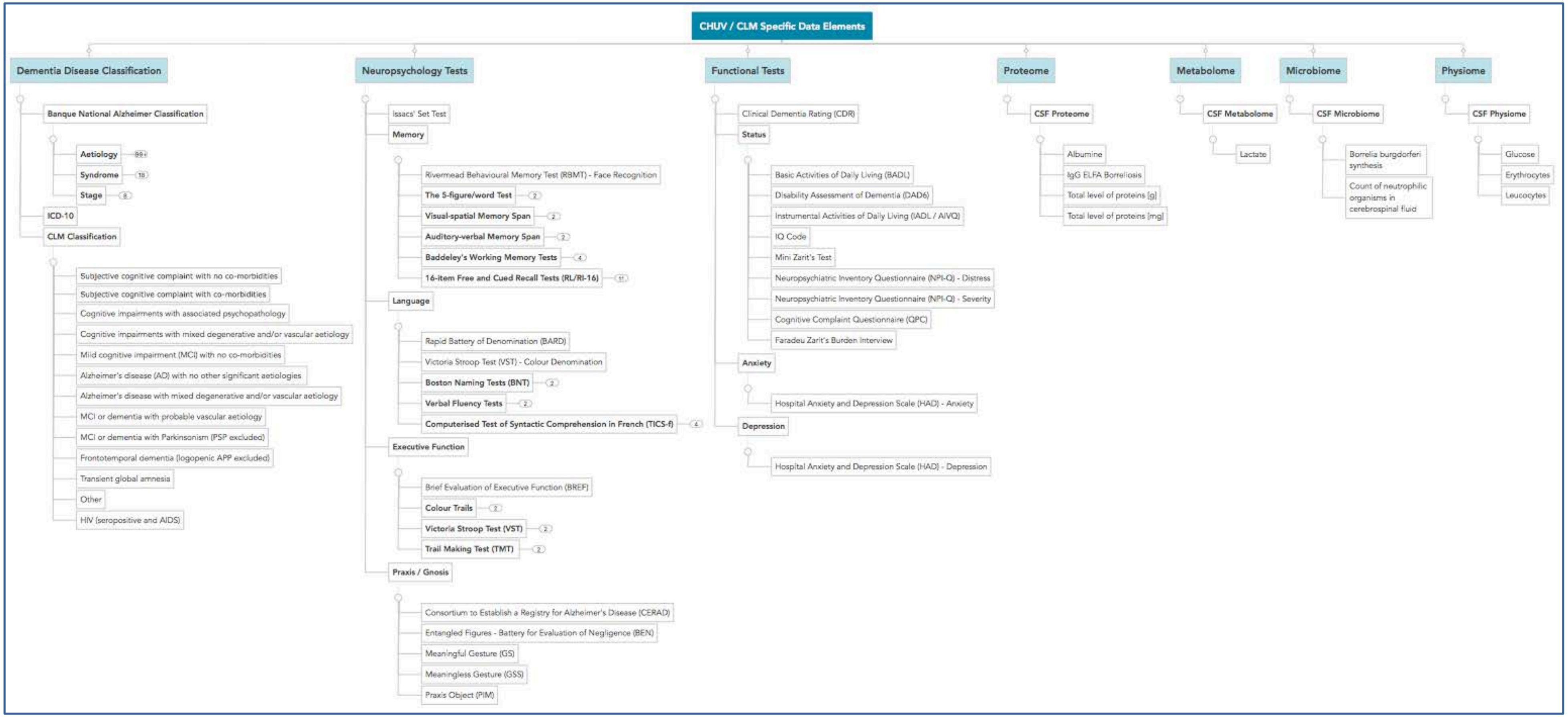

CHUV / CLM Specific Data Elements
Dementia Disease Classification
Banque National Alzheimer Classification
Aetiology
99+
Syndrome
18
Stage
8
ICD-10
CLM Classification
Subjective cognitive complaint with no co-morbidities
Subjective cognitive complaint with co-morbidities
Cognitive impairments with associated psychopathology
Cognitive impairments with mixed degenerative and/or vascular aetiology
Mild cognitive impairment (MCI) with no co-morbidities
Alzheimer's disease (AD) with no other significant aetiologies
Alzheimer's disease with mixed degenerative and/or vascular aetiology
MCI or dementia with probable vascular aetiology
MCI or dementia with Parkinsonism (PSP excluded)
Frontotemporal dementia (logopenic APP excluded)
Transient global amnesia
Other
HIV (seropositive and AIDS)
Neuropsychology Tests
Issacs' Set Test
Memory
Rivermead Behavioural Memory Test (RBMT) - Face Recognition
The 5-figure/word Test
2
Visual-spatial Memory Span
2
Auditory-verbal Memory Span
2
Baddeley's Working Memory Tests
4
16-item Free and Cued Recall Tests (RL/RI-16)
11
Language
Rapid Battery of Denomination (BARD)
Victoria Stroop Test (VST) - Colour Denomination
Boston Naming Tests (BNT)
2
Verbal Fluency Tests
2
Computerised Test of Syntactic Comprehension in French (TICS-f)
4
Executive Function
Brief Evaluation of Executive Function (BREF)
Colour Trails
2
Victoria Stroop Test (VST)
2
Trail Making Test (TMT)
2
Praxis / Gnosis
Consortium to Establish a Registry for Alzheimer's Disease (CERAD)
Entangled Figures - Battery for Evaluation of Negligence (BEN)
Meaningful Gesture (GS)
Meaningless Gesture (GSS)
Praxis Object (PIM)
Functional Tests
Clinical Dementia Rating (CDR)
Status
Basic Activities of Daily Living (BADL)
Disability Assessment of Dementia (DAD6)
Instrumental Activities of Daily Living (IADL / AIVQ)
IQ Code
Mini Zarit's Test
Neuropsychiatric Inventory Questionnaire (NPI-Q) - Distress
Neuropsychiatric Inventory Questionnaire (NPI-Q) - Severity
Cognitive Complaint Questionnaire (QPC)
Faradeu Zarit's Burden Interview
Anxiety
Hospital Anxiety and Depression Scale (HAD) - Anxiety
Depression
Hospital Anxiety and Depression Scale (HAD) - Depression
Proteome
CSF Proteome
Albumine
IgG ELFA Borreliosis
Total level of proteins [g]
Total level of proteins [mg]
Metabolome
CSF Metabolome
Lactate
Microbiome
CSF Microbiome
Borrelia burgdorferi synthesis
Count of neutrophilic organisms in cerebrospinal fluid
Physiome
CSF Physiome
Glucose
Erythrocytes
Leucocytes

Figure 49: CLM/CHUV Lausanne Specific Data Taxonomy

7. Technology Readiness Level Assessment

The Medical Informatics Platform is a sophisticated software system developed out of many individual technologies (i.e. components) integrated into a complex functional solution for descriptive and predictive analysis of patient datasets, including the combination of data originating from their hospital health records and processed brain scans.

7.1 Adaptation of the standard EC TRL scale

HBP adaptations of the source EC TRL scale addresses the aspects of research solutions that will need integration of various technologies, interaction with users and validation of the systems in user environments. These adaptations are essential for the comprehensive evaluation of the technological maturity of the Medical Informatics Platform, because its technological value and value for users depend on the maturity of the fully integrated and operational system. The platform is a complex solution developed out of a number of individual technologies/components.

The Medical Informatics Platform is a data-intensive analytics solution. It uses available data (patient biomedical and other health-relevant information from their hospital medical records and neuroimaging data from brain scans) to produce more data (the results of the descriptive and predictive data analysis). The technological maturity of such a solution, and its value for the users is a function of the quality of new data (or knowledge) production, i.e. it is a function of the quality of the data analysis results. The quality of the data analytics ultimately depends on the type, quality, variability and volume of the analysed data.

Therefore, the technological maturity of the Medical Informatics Platform and its value for the users directly depends on the number and variety of participating hospitals and the number and type of datasets that are available to the Platform for analytics. The number and variety of participating hospitals and research institutes will not only depend on the technological maturity of the Platform. Financial and organisational aspects determine equally much the success of the widespread deployment of the solution.

For an accurate evaluation of the MIP technological maturity and a precise communication of the technology readiness level in any of the project stages, it is crucial to take the following aspects into account:

- The TRL setback mechanisms need to be incorporated, as their exclusion would mean that when (not if!) they occur, funding of specific activities would be (temporarily) stopped, leading to an unnecessary destruction of capital. In contrast to the implicit linear character of both EC TRL scale and its HBP adaptation, feedback models show that research is needed even at the higher TRL levels, i.e. an increase in maturity also requires additional R&D. The implication is that in every stage certain kinds of R&D should be incorporated

- Innovation is usually built up from different technologies. Therefore the TRL scaling should make a distinction between R&D on individual technologies, integration of those technologies and pilot production. Most of the relevant aspects are provided for the HBP adaptation of the EC TRL scale, but the focus of the higher TRLs seem to be on the “small number of users”. In the case of the Medical Informatics Platform, the TRL scaling should account for the wide deployment and the maturity of the corresponding technologies. The software “manufacturing” technologies needed (CI/CD, system monitoring, O&M tools, version control, operation processes, etc.), can be seen as just another set of technologies

- Innovation is not about technology (product and process) alone. Financial and organisational activities can be crucial to commercial success. Both the EC TRL scale and its HBP adaptation are clearly about product oriented technologies. Their focus is apparently on product development, but very little on the ability of the production on a broader scale and there is no explicit mentioning of organisational requirements. Non-technological aspects, the readiness of an organisation to implement the innovation, for example, should be incorporated into the TRL definitions. For example, the development of accompanying services, including tools, processes and organisation, is just one example that is crucial in case of the Medical Informatics Platform, as it determines the success and sustainability of the wide-spread platform deployment

An integrative TRL assessment approach, combining different technologies and addressing market and organisational issues, is recommended to assess and communicate the MIP technology readiness level. We have decided to adopt the recommendation of European Association of Research and Technology Organisations (EARTO) for its close compliance to the needs of the SP8 strategic objectives and for the nature of the Platform. The different maturity stages are summarised in Figure 50 and details are provided in Table 13: MIP TRL definition overview table.

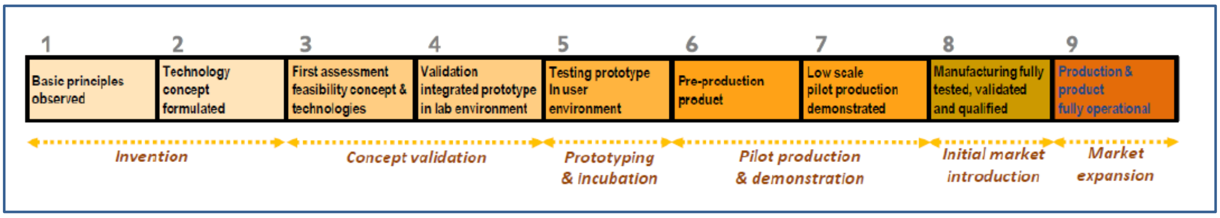

1
Basic principles observed
2
Technology concept formulated
3
First assessment feasibility concept & technologies
4
Validation integrated prototype in lab environment
5
Testing prototype In user environment
6
Pre-production product
7
Low scale pilot production demonstrated
8
Manufacturing fully tested, validated and qualified
9
Production & product fully operational
Invention
Concept validation
Prototyping & incubation
Pilot production & demonstration
Initial market introduction
Market expansion

Figure 50: The adaptation of EC TRL scale to the SP8-MIP needs

Finally, an interaction between disciplines, a trans-disciplinary and user-centric approach is needed to solve societal challenges by connecting various technologies, connecting one technology to multiple applications, connecting technologies to non-technological disciplines allowing to take users perspective into account as well as to look at solutions bridging commercial interests and society needs. These aspects are all relevant and essential criteria for the assessment of a complex data intensive solution, such as the Medical Informatics Platform.

Furthermore, the successful wide-scale use of the new technologies inevitably changes the perceptions and needs of the users and society as a whole. As a consequence of the ever-lasting evolution of the user needs, an organisation needs to provide for the sustainable phased development of the solutions. In a typical mature R&D organisation, while phase N of the complex solution is deployed and in operation (TRL9 level), product phase N+1 is in the TR5-7 developmental stage, phase N+2 is in the conceptualisation and prototyping stage at TR level 2-4, and the product phase N+3 can be in the early research stage at TRL1 level.

Table 13: MIP TRL definition overview table

Cluster	TRL	HBP Terminology	MIP/EARTO Reading	MIP Definition and Description
Invention	TRL1	Project Initiation	Basic principles observed	Basic scientific research is translated into potential new basic principles that can be used in new technologies
	TRL2	Conceptualisation	Technology concept formulated	Potential application of the basic (technological) principles is identified, including their technological concept. Also, the first wide-scale software deployment principles are exploited, as well as possible markets identified. A small research team is established to facilitate assessment of technological feasibility
Concept validation	TRL3	Proof of Concept Implementation	First assessment of feasibility of the concept and technologies	Based on the preliminary study, an analysis is conducted to assess technical and market feasibility of the concept. This includes active R&D on a laboratory scale and first discussions with potential clients from major European university hospitals. The research team is further expanded and an early market feasibility assessed
	TRL4	Prototype Component	Validation of integrated prototype in a laboratory	Basic technological components are integrated to assess early feasibility by testing in a laboratory environment. Wide-scale software deployment is actively researched and analysed, identifying main production principles. Lead hospitals and institutes are engaged to ensure connection with demand. Organisation is prepared to enter into scale up, possible services prepared and full market analysis conducted
Prototyping and incubation	TRL5	Prototype Integration	Testing of the prototype in a user environment	The system is tested in a user environment, connected to the broader technological infrastructure. Actual use is tested and validated. Wide-scale deployment is prepared and tested in a laboratory environment and lead hospitals and institutes can test pre-production products. First activities within the organisation are established to further scale up to pilot production and marketing
Pilot production and demonstration	TRL6	Prototype-to-Real-world Integration	Pre-production of the product, including testing in a user environment	Product and manufacturing technologies are now fully integrated in a pilot line or pilot plant (low-rate software deployment). The interaction between the product and wide-scale software deployment technologies are assessed and fine-tuned, including additional R&D. Lead hospitals and institutes test the early products and wide-scale software deployment process and the organisation of production is made operational (including marketing, logistics, production and others)

Cluster	TRL	HBP Terminology	MIP/EARTO Reading	MIP Definition and Description
	TRL7	Operational Integration	Low-scale pilot production demonstrated	Wide-scale software deployment process is now fully operational at a low rate, producing actual final developed products. Lead hospitals and institutes test these final products and organisational implementation is finalised (full marketing established, as well as all other production activities fully organised). The product is formally launched into first early adopter hospitals and institutes
Initial market introduction	TRL8	Deployment	Wide-scale software deployment process fully tested, validated and qualified	Wide-scale software deployment of the product and the final version of the product are now full established, as well as the organisation of production and marketing. Full-launch of the product is now established at the European markets
Market expansion	TRL9	Production	Production and product fully operational and competitive	Full production is sustained, the product is expanded to worldwide markets and incremental changes of the product create new versions. Wide-scale software deployment and overall production is optimised by continuous incremental innovations of the process. Worldwide markets are fully addressed

7.2 Integrated system technology readiness level assessment

As discussed in the previous sub-chapter, for the precise assessment of the MIP's TRL at the end of the SGA1 project phase, and for full compliance with the plans for the technology maturation as defined in the SP8 SGA2 proposal, we decided to adopt the EARTO adaptation of the EC TRL definitions (see Table 13: MIP TRL definition overview table).

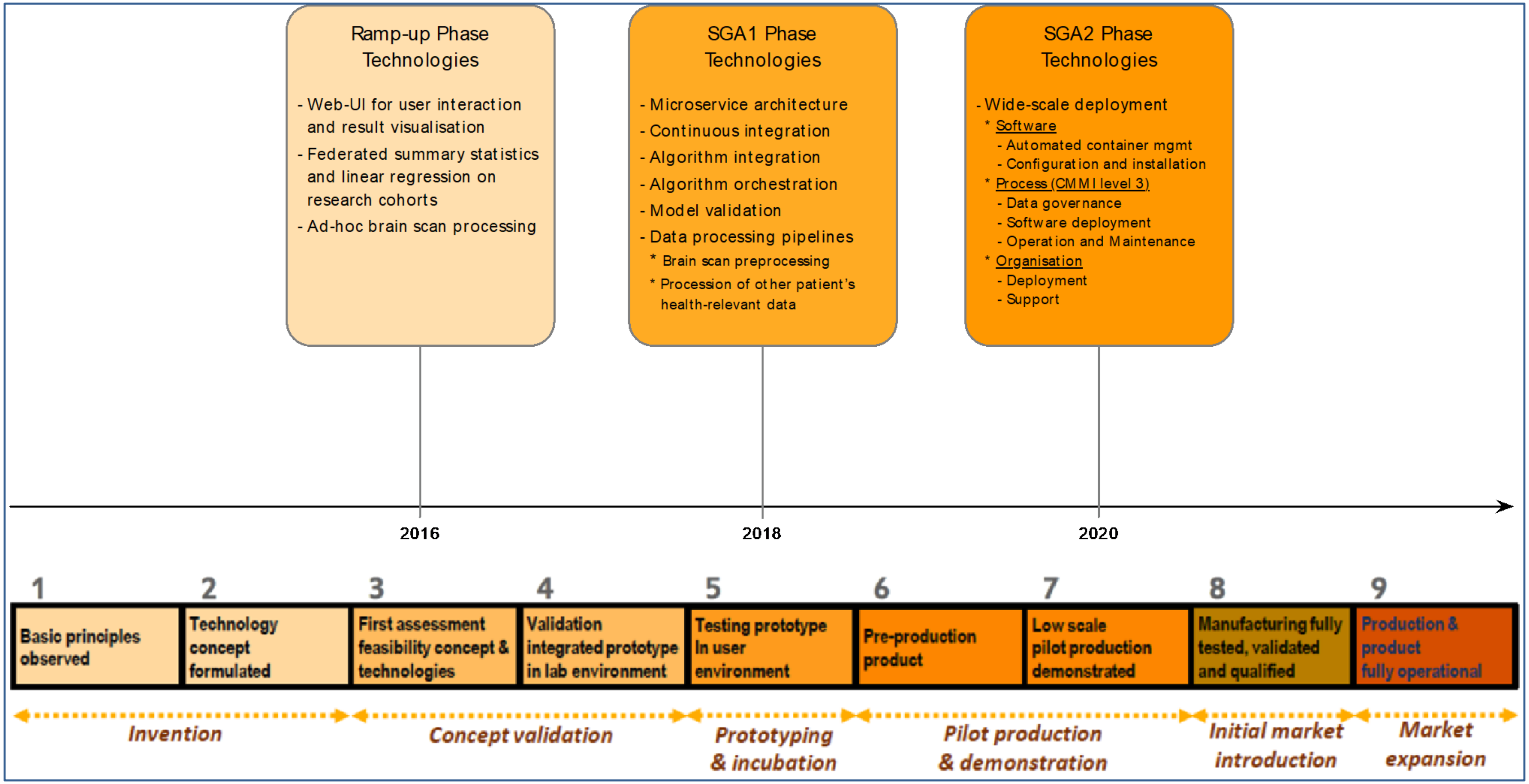
The MIP is a data-intensive solution. MIP of a higher level of technological maturity requires access to big data for technologically more advanced ways to discover biological signatures of diseases by applying predictive machine learning and deep learning algorithms. The emphasis is therefore also on the development of a mature wide-scale production technology of the Platform, with the corresponding processes and organisational aspects as prerequisites for its wide-scale deployment to get access to more patient datasets.

Table 14: Technology readiness level assessment of the key technologies / components

ID	Component Name	TRL	Component Type	Description
2938	Algorithm Orchestrator	TRL5	SOFTWARE	The component is integrated into the MIP ecosystem and tested in a user environment.
647	Algorithm Repository	TRL5	SOFTWARE	The component is integrated into the MIP ecosystem and tested in a user environment
645	Model Benchmark and Validation	TRL5	SOFTWARE	The component is integrated into the MIP ecosystem and tested in a user environment
646	Predictive Disease Models	TRL5	SOFTWARE	The component is integrated into the MIP ecosystem and tested in a user environment
633	Portal DB (Articles, Experiments, Models)	TRL5	SOFTWARE	The component is integrated into the MIP ecosystem and tested in a user environment
1595	Distributed Query Processing Engine – Master	TRL4	SOFTWARE	The component is integrated into the MIP ecosystem to assess early feasibility by testing in a laboratory environment
1596	Distributed Query Processing Engine – Worker	TRL4	SOFTWARE	The component is integrated into the MIP ecosystem to assess early feasibility by testing in a laboratory environment
638	Query Engine	TRL4	SOFTWARE	The component is integrated into the MIP ecosystem to assess early feasibility by testing in a laboratory environment

ID	Component Name	TRL	Component Type	Description
687	Data Governance Methodology	TRL3	service	Based on the preliminary study, technical and market feasibility of the concept is analysed and discussed with potential clients from major European hospitals
587	Data Mapping and Transformation Specification	TRL3	data	Based on the preliminary study, technical and market feasibility of the concept is analysed and discussed with potential clients from major European hospitals and tested in one hospital (CHRU Lille)
1580	Online Data Integration Module	TRL4	software	The component is integrated into the MIP ecosystem to assess early feasibility by testing in a laboratory environment
671	Neuromorphometric Processing	TRL7	software	Lead hospitals and institutes are using the solution. The component is formally launched and training is established. The solution is developed and managed in an academic organisation. Product manufaturing and marketing organisation needed for TRL8 categorisation is not established
664	Airflow DAGs	TRL4	software	The component is integrated into the MIP ecosystem and tested in a laboratory environment. It is based on the TRL8/9 categorised Apache Airflow solution
2927	Data Catalogue	TRL4	software	The component is integrated into the MIP ecosystem and tested in a laboratory environment. It is based on the TRL9 PostgreSQL DBMS
2926	Data Capture Database	TRL4	software	The component is integrated into the MIP ecosystem and tested in a laboratory environment. It is based on the TRL9 PostgreSQL DBMS and the star database schema is compatible with TRL7 I2B2 solution
669	Common Data Elements Database	TRL4	software	The component is integrated into the MIP ecosystem and tested in a laboratory environment. It is based on the TRL9 PostgreSQL DBMS and the star database schema is compatible with TRL7 I2B2 solution

ID	Component Name	TRL	Component Type	Description
102	MIP Microservice Infrastructure	TRL5	software	The component is integrated into the MIP ecosystem and tested in a user environment
2940	Data De-identifier	TRL5	software	The component is integrated into the MIP ecosystem and tested in a user environment.
2936	MIP De-identification Profiles	TRL5	model	The component is integrated into the MIP ecosystem and tested in a user environment.
2935	MIP De-identification Strategy	TRL5	report	The component is integrated into the MIP ecosystem and tested in a user environment.



Ramp-up Phase Technologies

- Web-UI for user interaction and result visualisation
- Federated summary statistics and linear regression on research cohorts
- Ad-hoc brain scan processing

SGA1 Phase Technologies

- Microservice architecture
- Continuous integration
- Algorithm integration
- Algorithm orchestration
- Model validation
- Data processing pipelines
  - * Brain scan preprocessing
  - * Procession of other patient's health-relevant data

SGA2 Phase Technologies

- Wide-scale deployment
  - * Software
    - Automated container mgmt
    - Configuration and installation
  - * Process (CMMI level 3)
    - Data governance
    - Software deployment
    - Operation and Maintenance
  - * Organisation
    - Deployment
    - Support

1 Basic principles observed

2 Technology concept formulated

3 First assessment feasibility concept & technologies

4 Validation integrated prototype in lab environment

5 Testing prototype In user environment

6 Pre-production product

7 Low scale pilot production demonstrated

8 Manufacturing fully tested, validated and qualified

9 Production & product fully operational

Invention · *Concept validation* · *Prototyping & incubation* · *Pilot production & demonstration* · *Initial market introduction* · *Market expansion*

Figure 51: Transition of MIP technology readiness level and future roadmap